

fig. 1

FIGURE 2 IN SECTION A-A

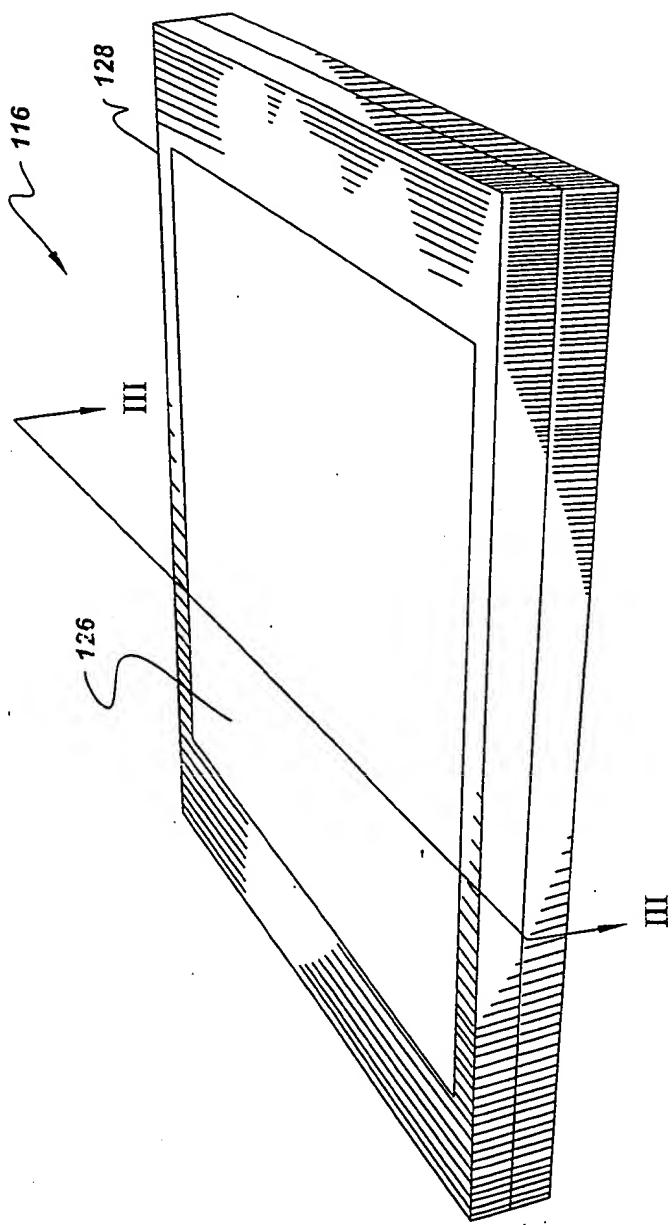


fig. 2
(PRIOR ART)

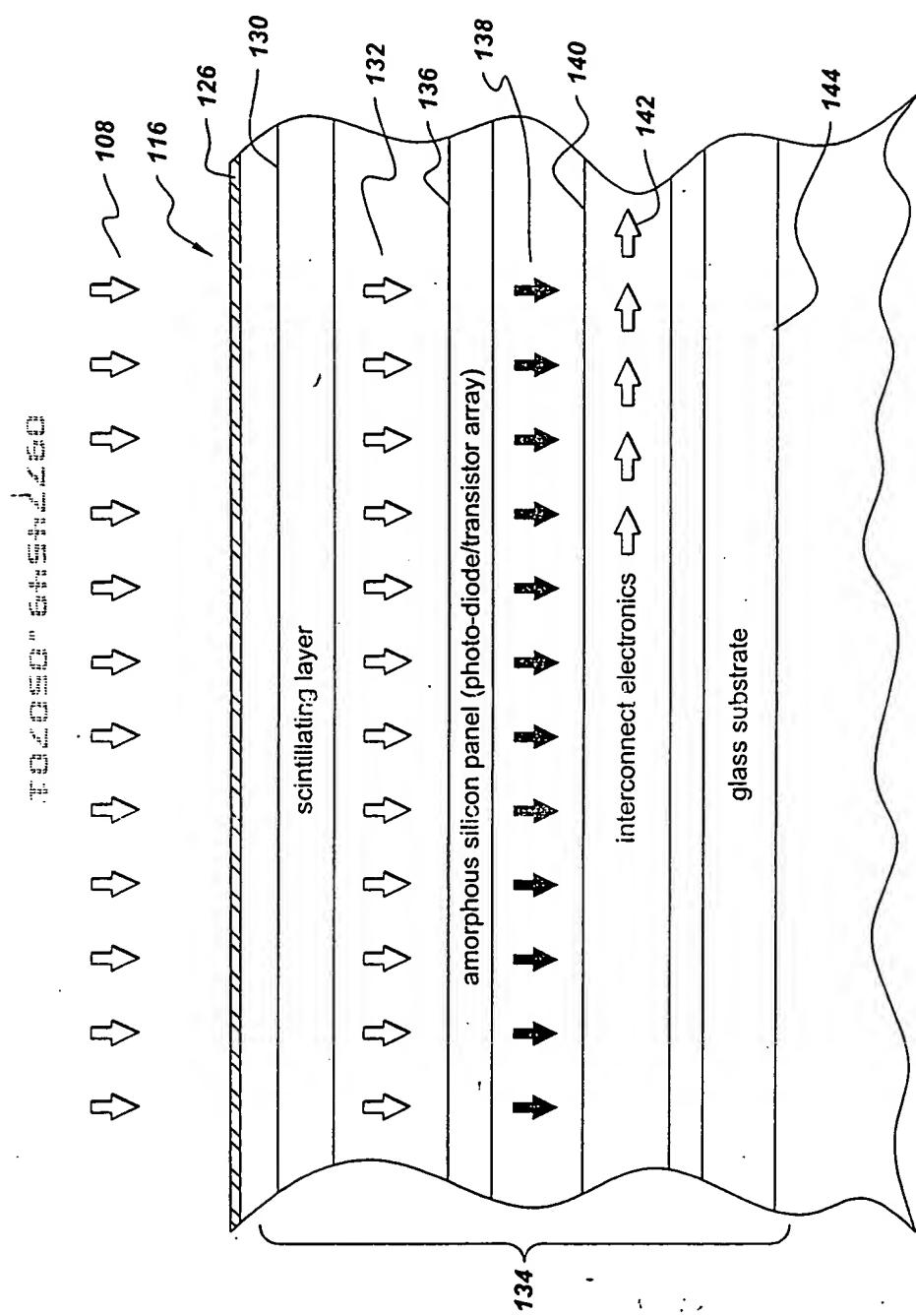
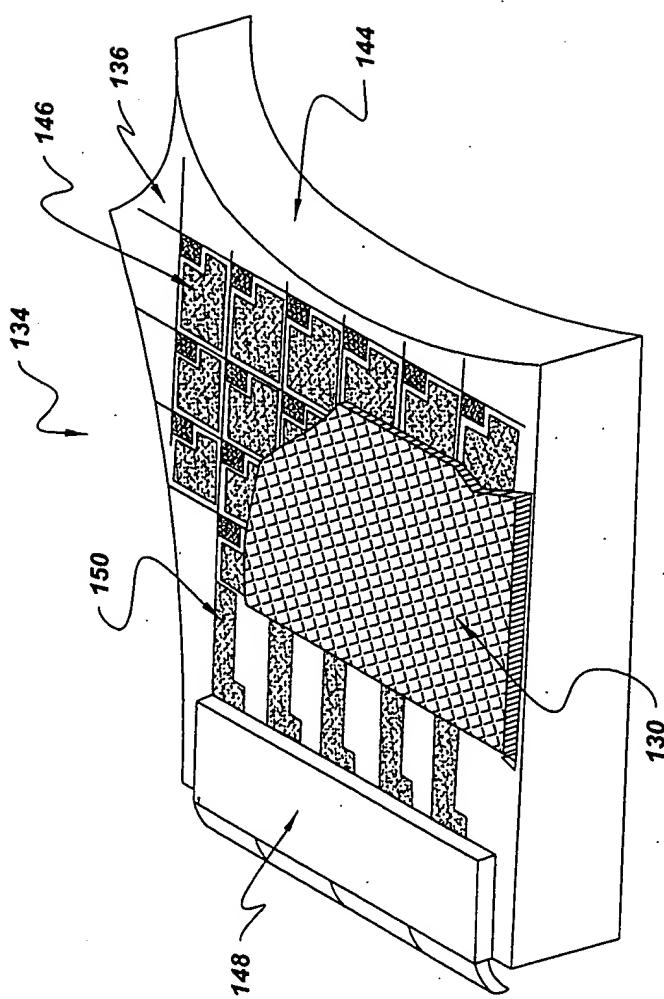


fig. 3

fig. 4
(Prior Art)



102550 "prior art"

702555-5451250

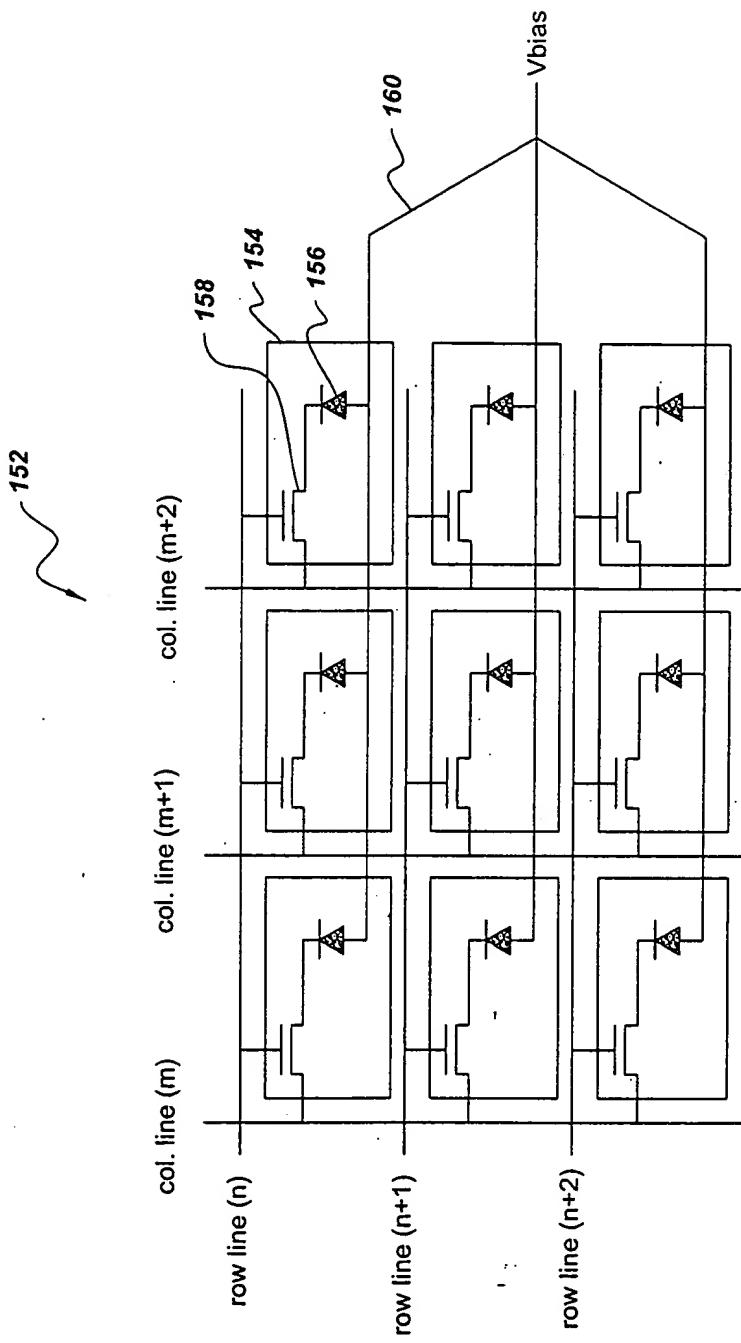


fig. 5
(Prior Art)

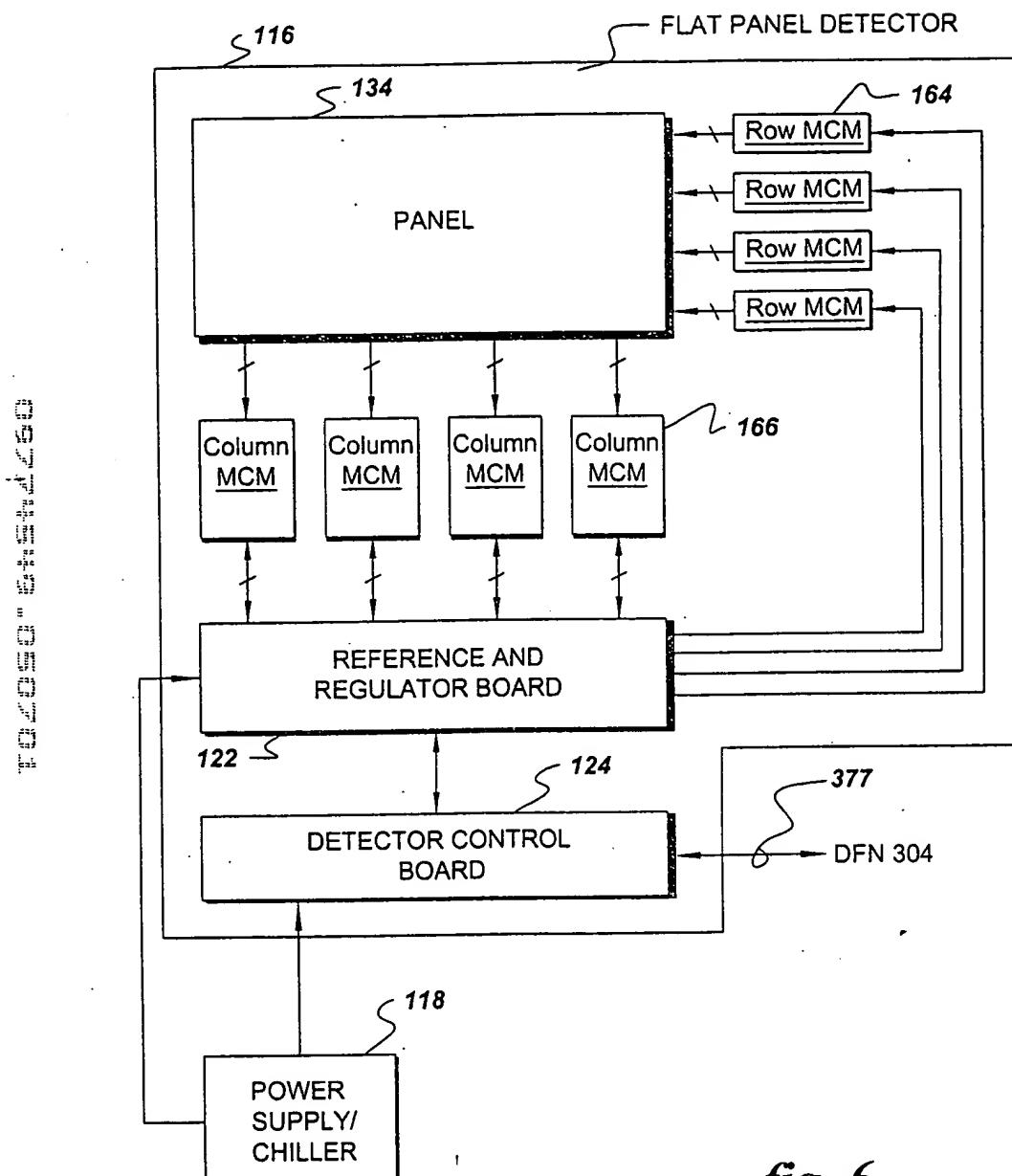


fig. 6
(PRIOR ART)

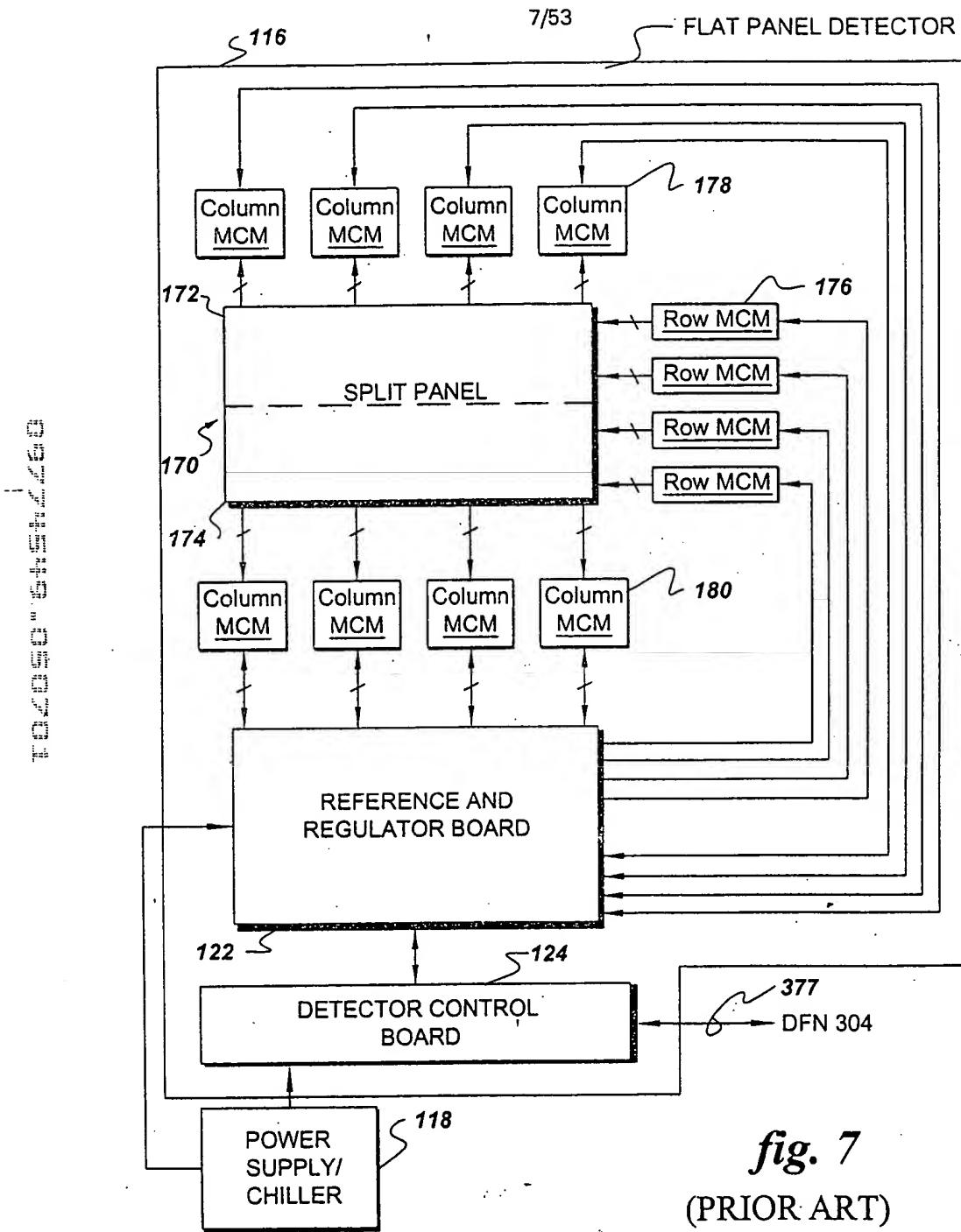
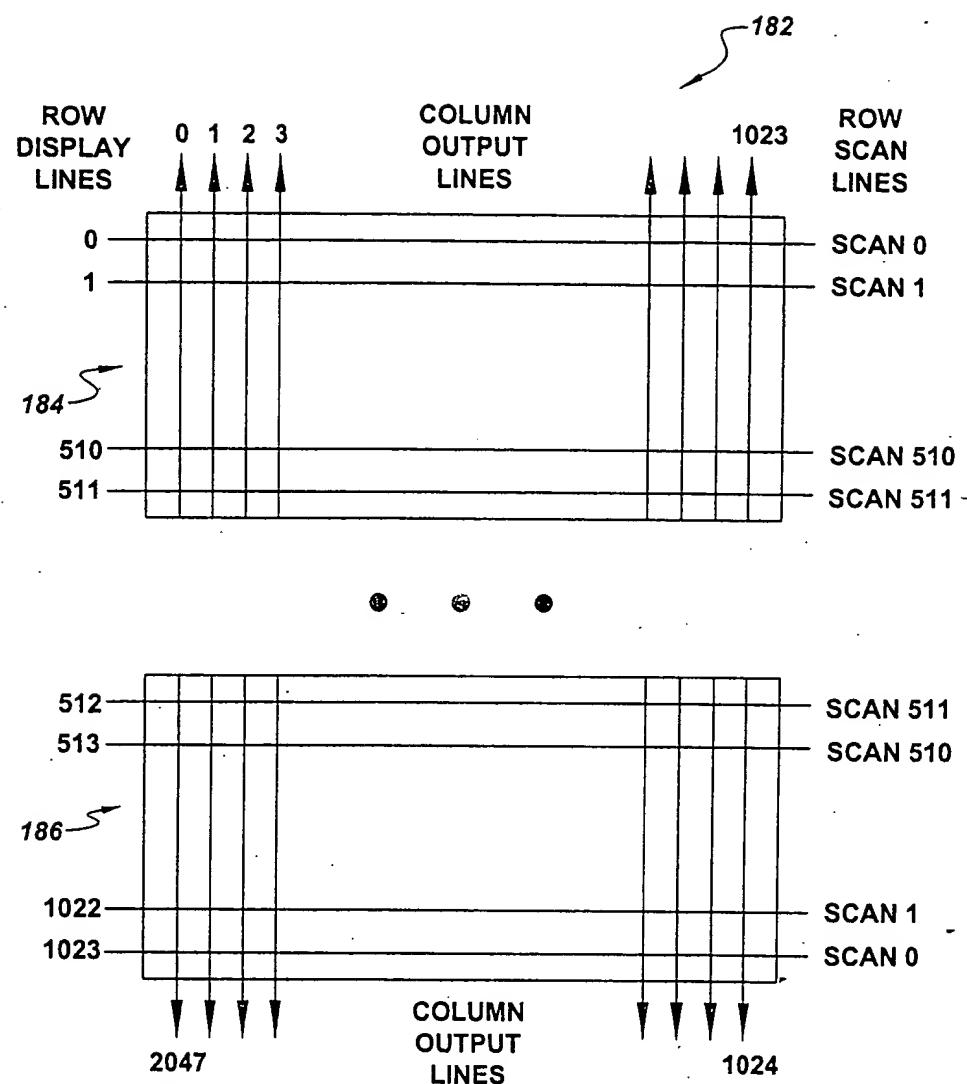


fig. 7
(PRIOR ART)



CARDIAC/SURGICAL DIGITAL X-RAY PANEL

fig. 8
(PRIOR ART)

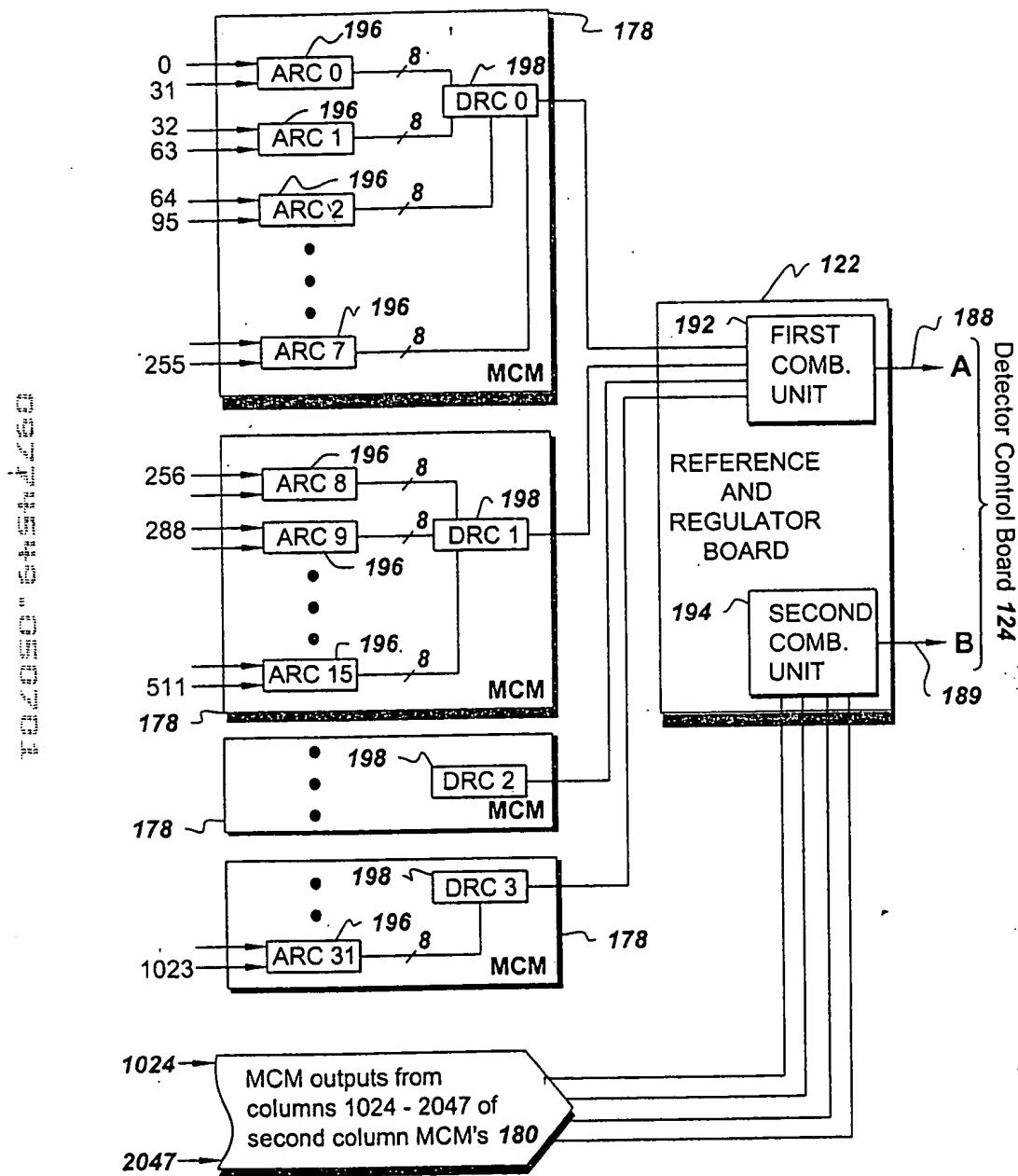
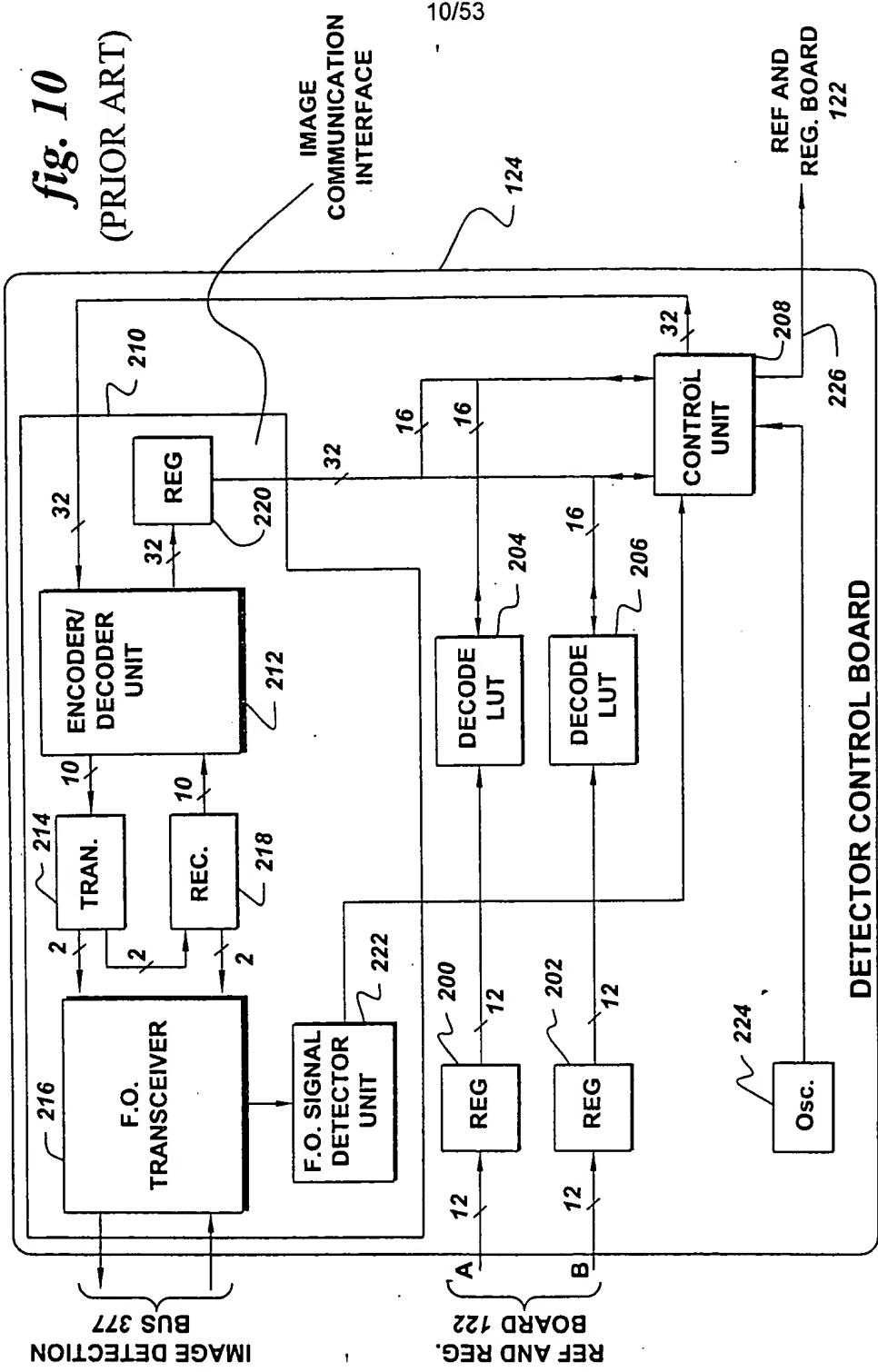
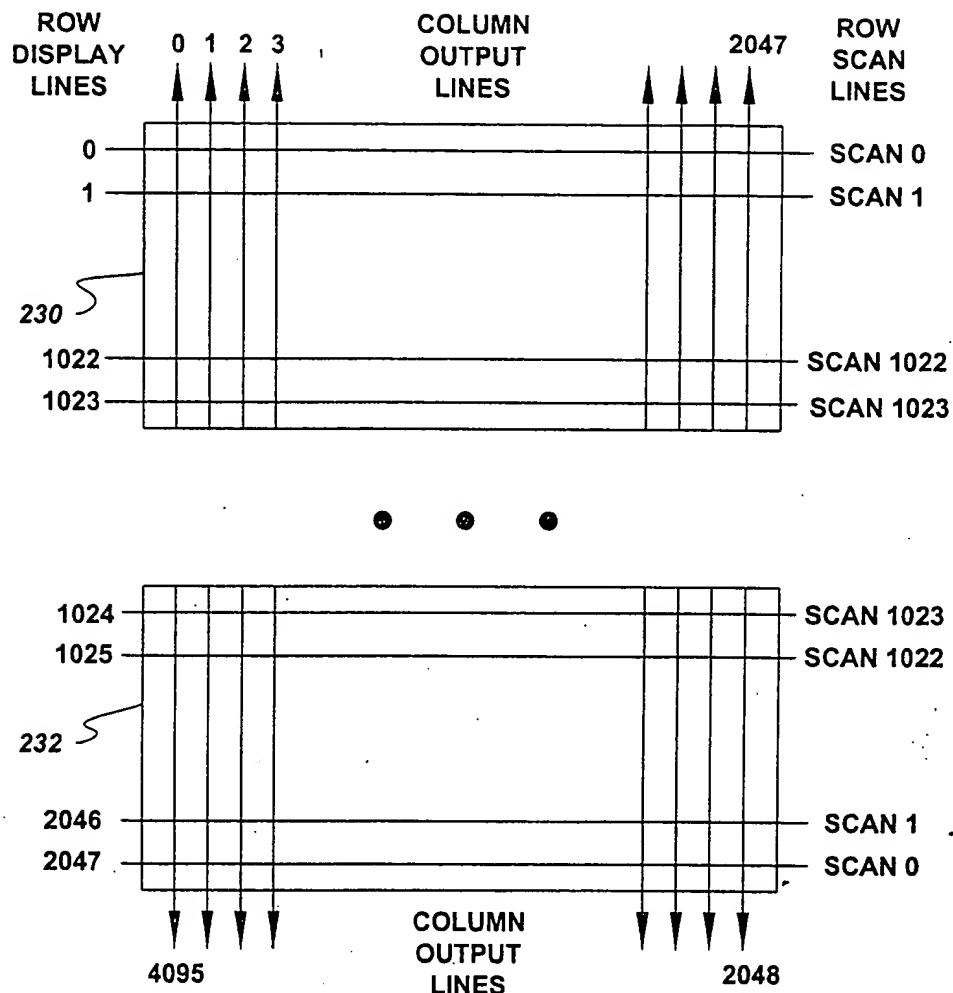


fig. 9 (PRIOR ART)

fig. 10
(PRIOR ART)



228



RADIOGRAPHY DIGITAL X-RAY PANEL

fig. 11 (PRIOR ART)

TOP SECRET//SI//EYESAFE

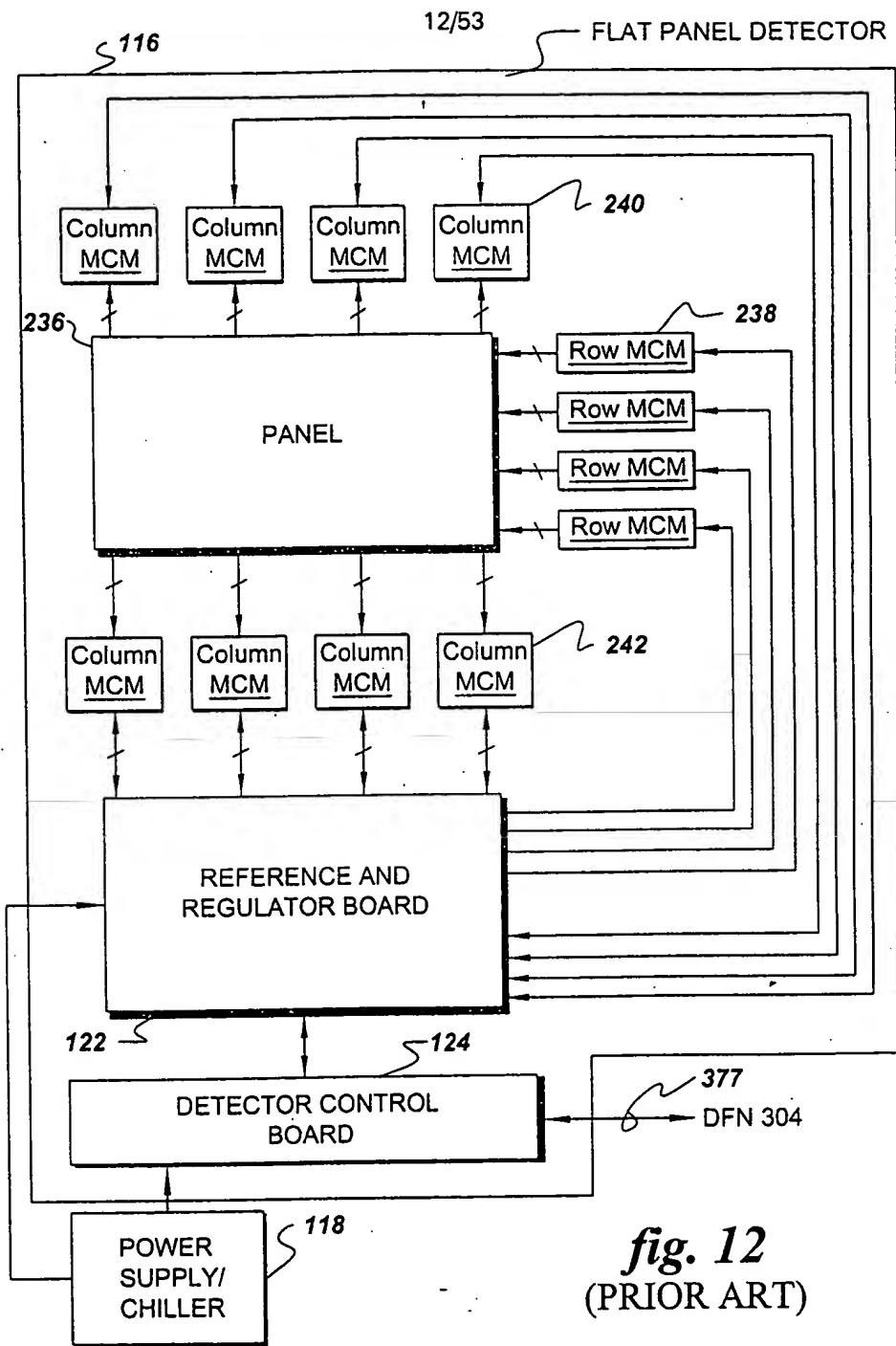
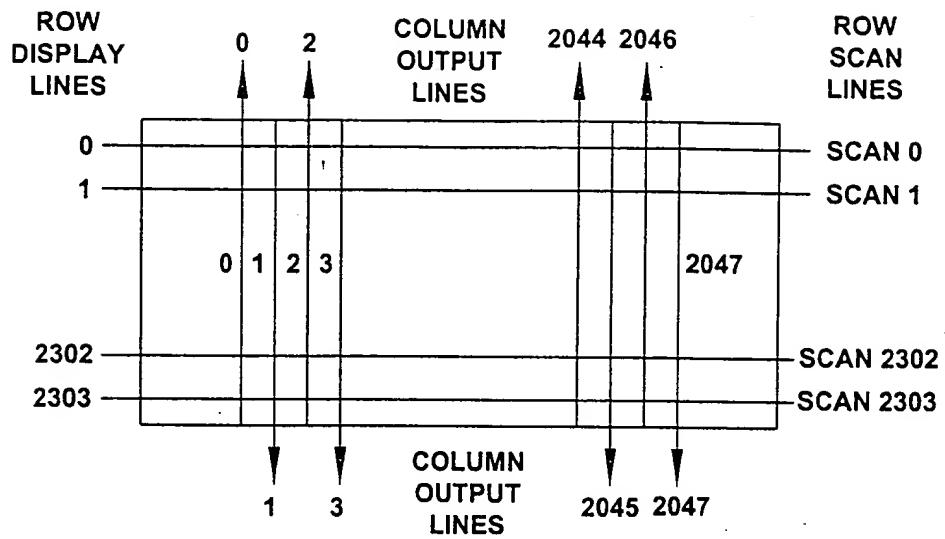


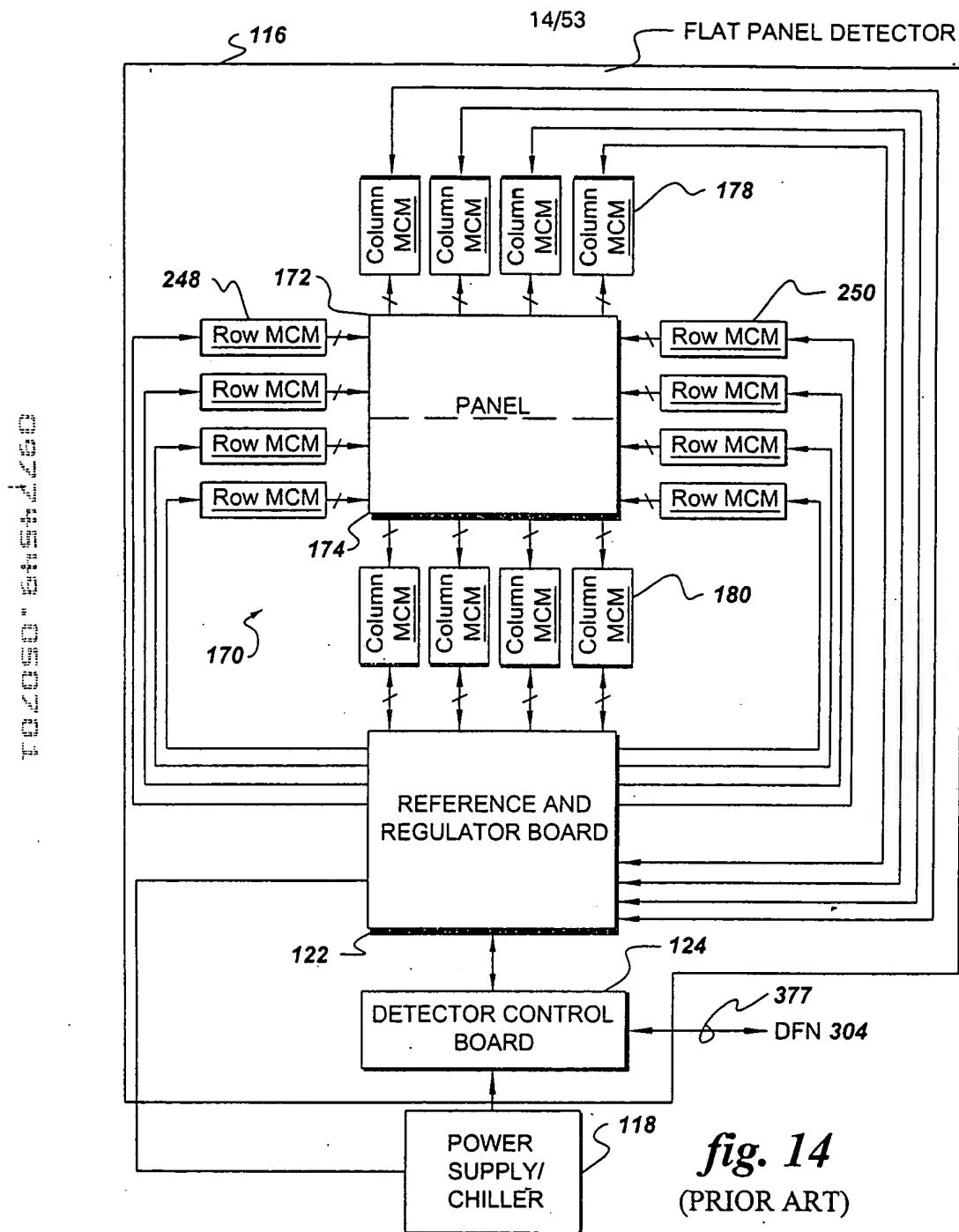
fig. 12
(PRIOR ART)

244



MAMMOGRAPHY DIGITAL X-RAY PANEL

fig. 13
(PRIOR ART)



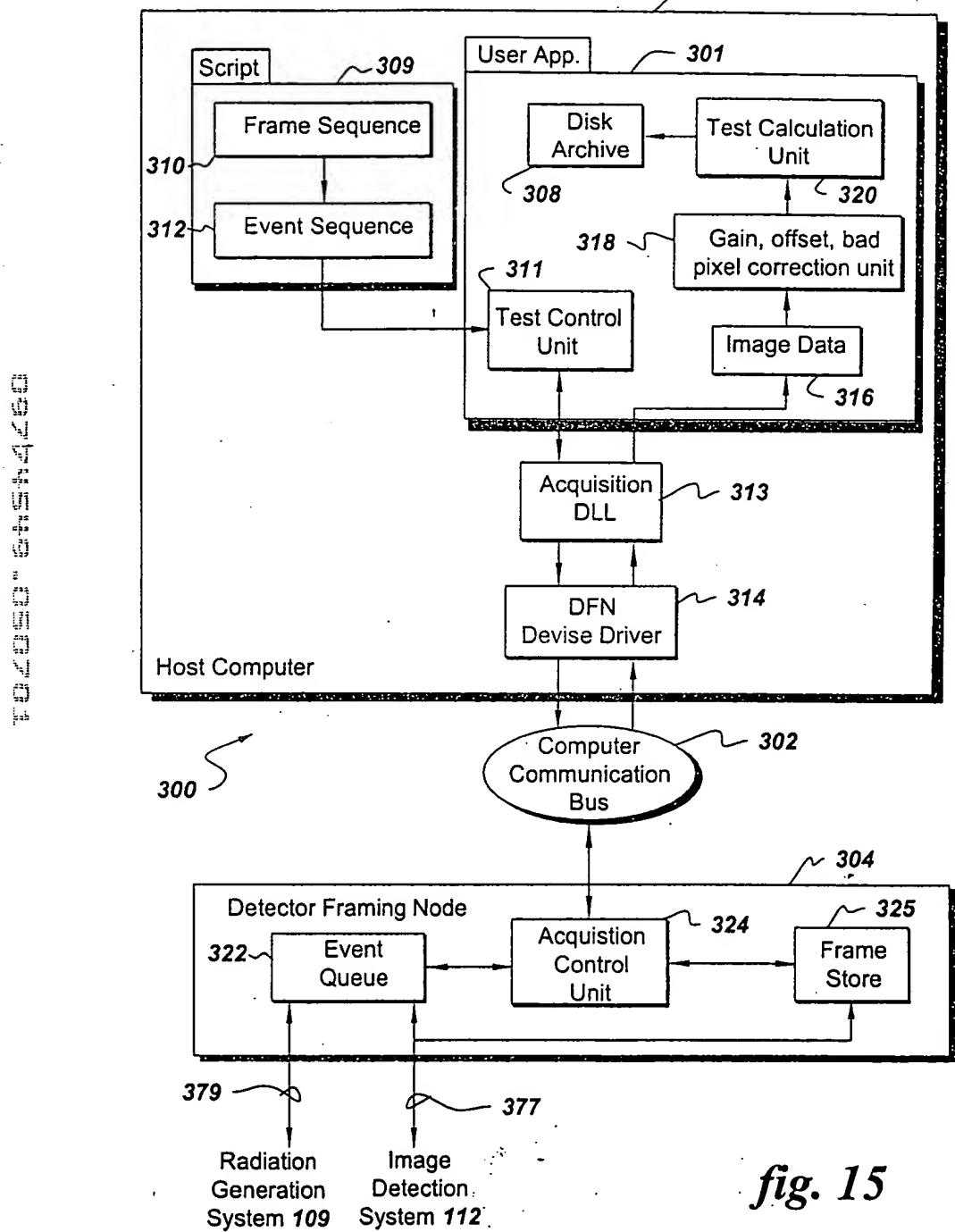


fig. 15

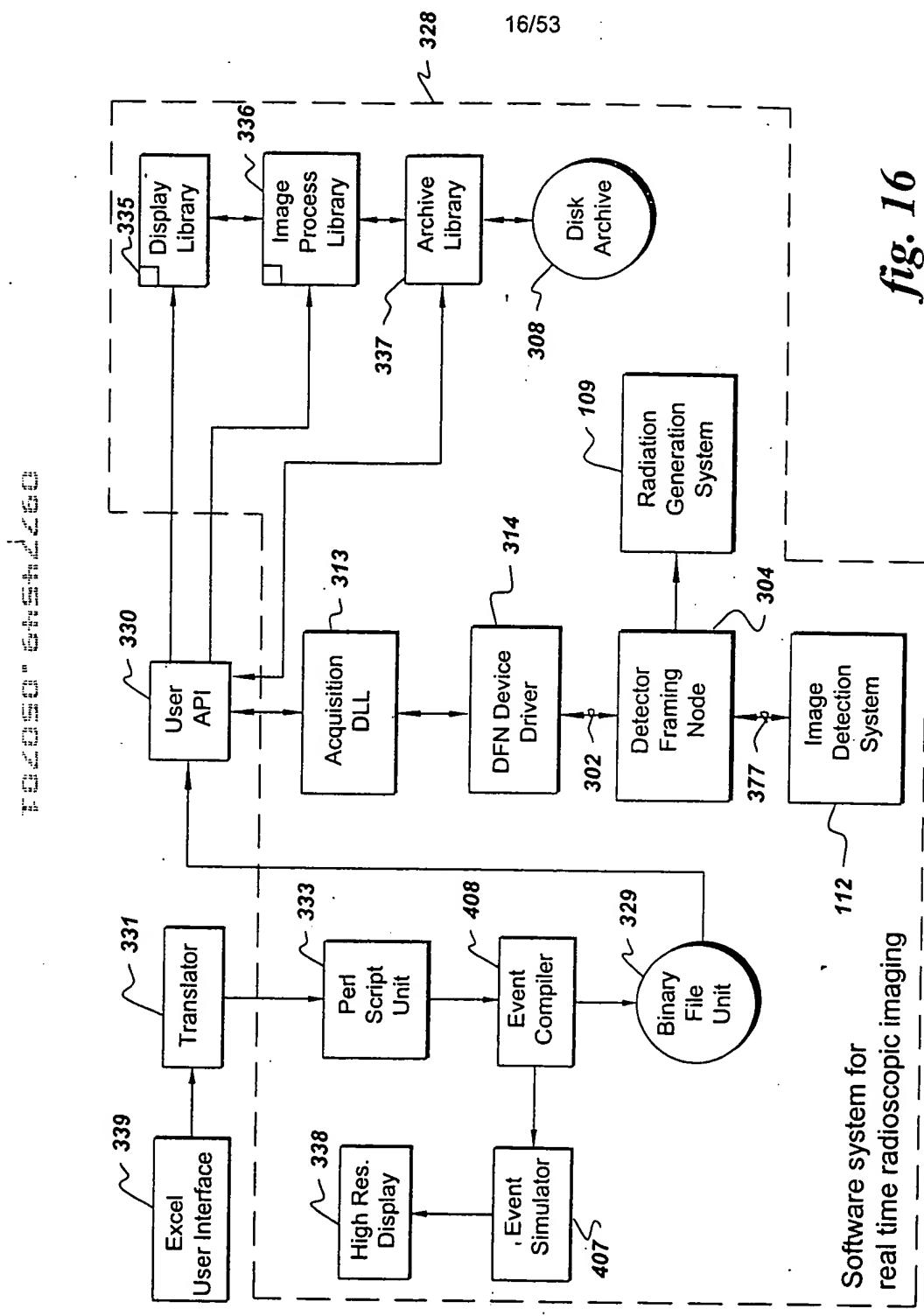
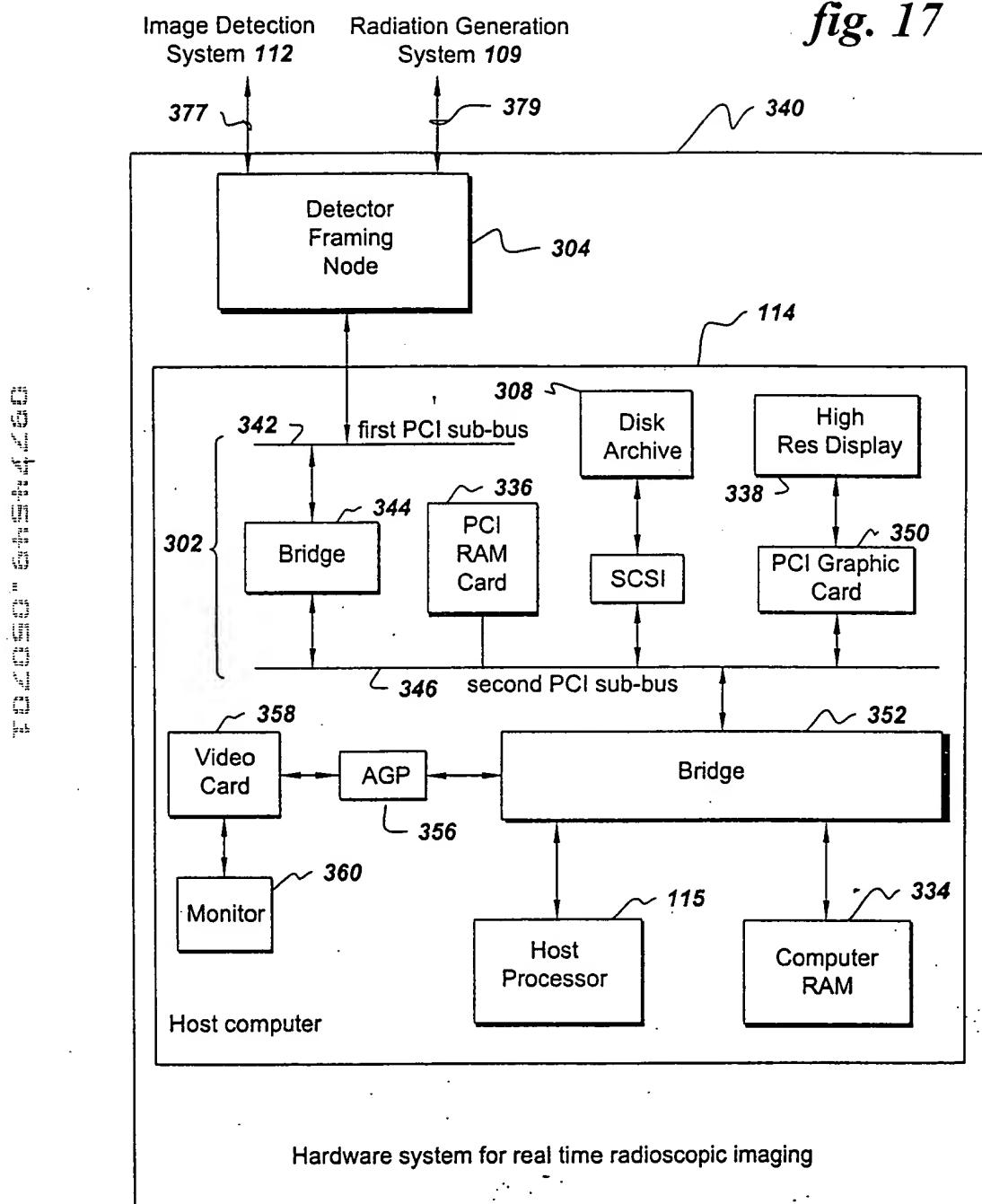


fig. 16

fig. 17



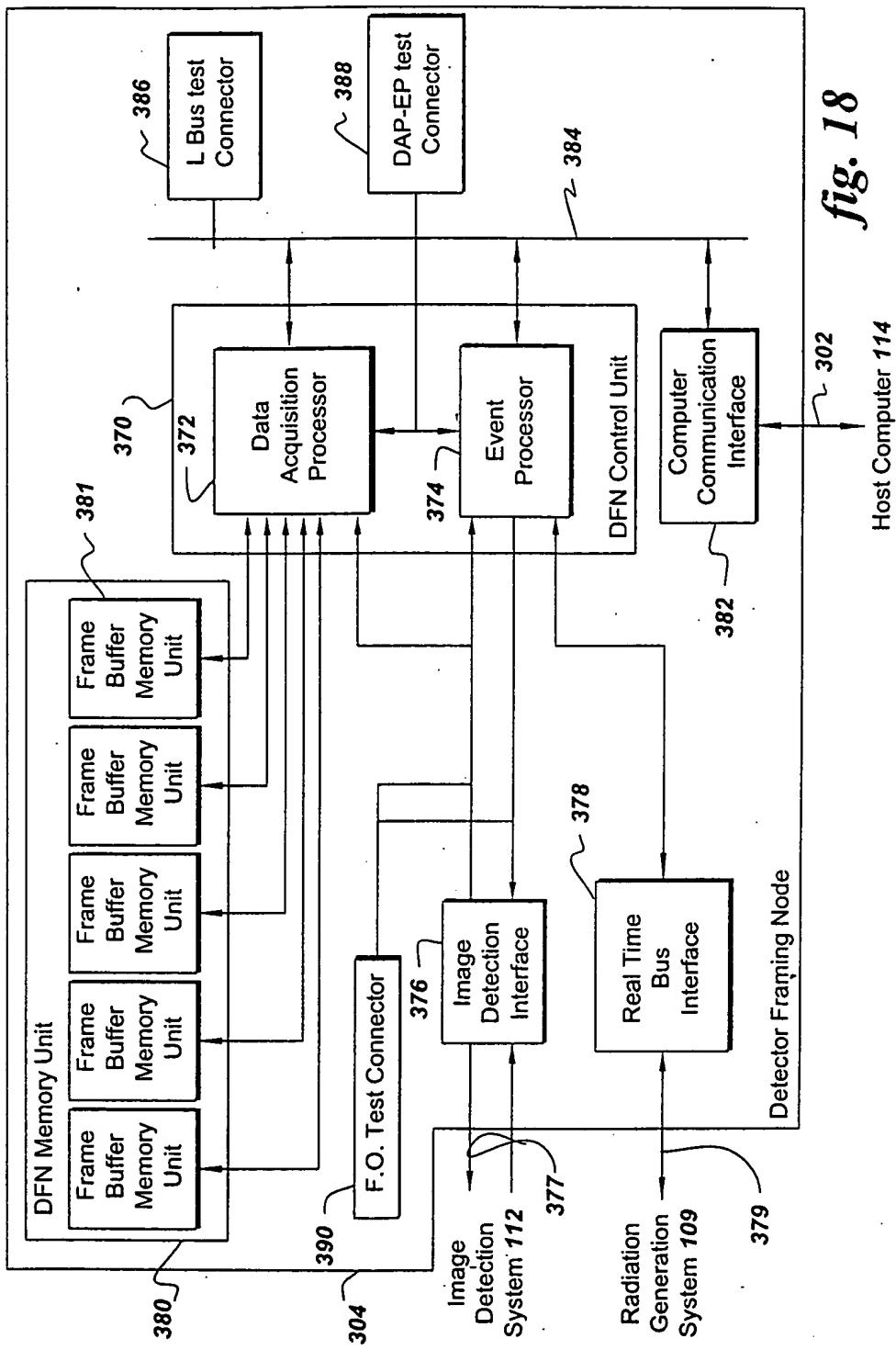


fig. 18

Host Computer 114

Panel Setup	Real Time	30	unlimited	<5 frames	host	none	gbr
Single Frame	Post Process	-	-	Delay ~.1 sec	"	y	
Single Frame	Post Process	-	-	Delay ~.2 sec	"	y	y
Real Time	Real Time	R	unlimited	<5 frames	host	none	
Real Time	Real Time	R-X	unlimited	<5 frames	"	y	
Real Time	Real Time	R-Y	unlimited	<5 frames	"	y	y

fig. 19

Modality	image size	Frames Stored host memory
Cardiac	1024 X 1024	200
Rad	2048 X 2048	50
Mammo	2304 X 2048	44

fig. 20

102000-000044260

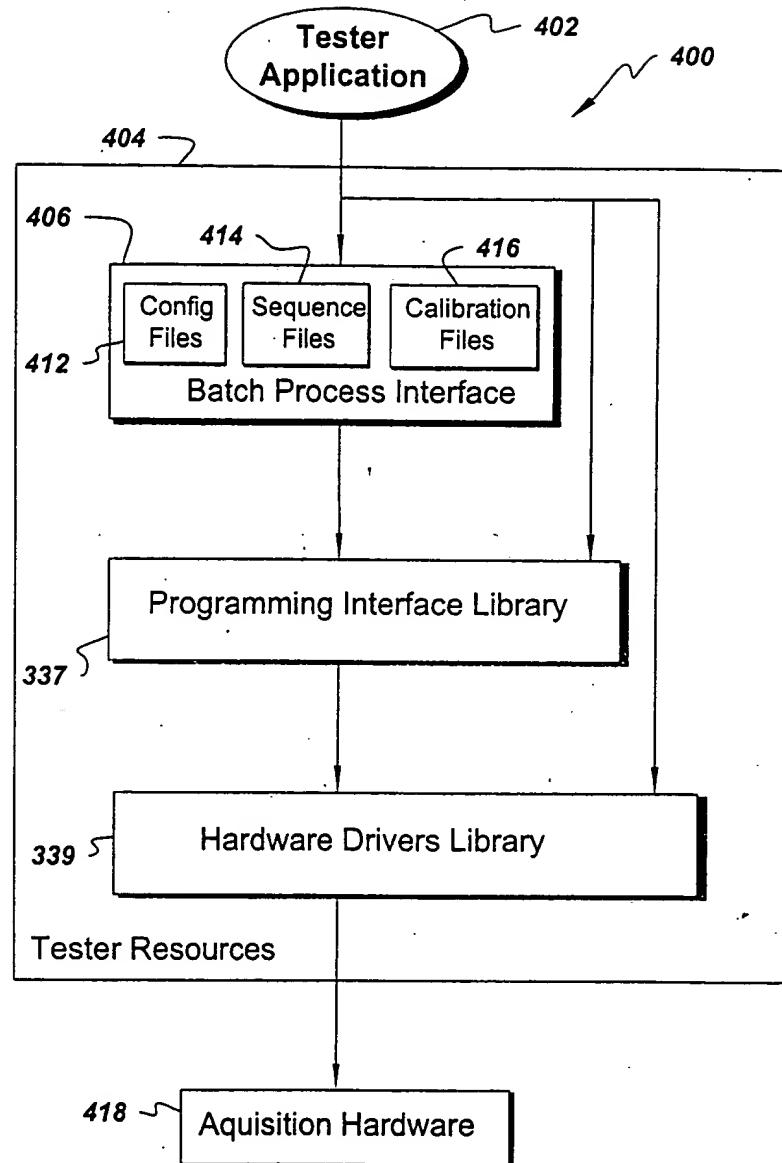


fig. 21

100-00500-034544250

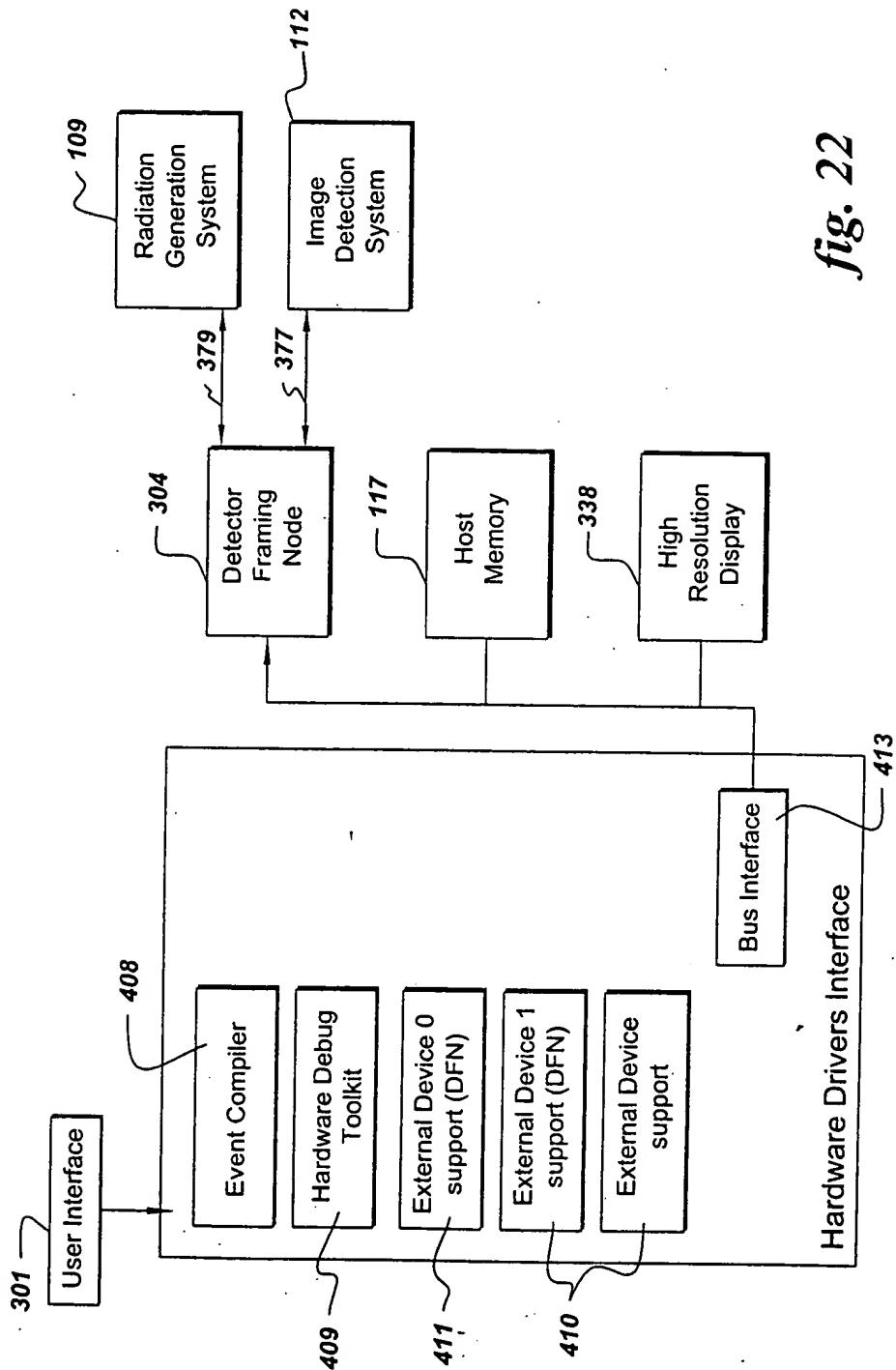


fig. 22

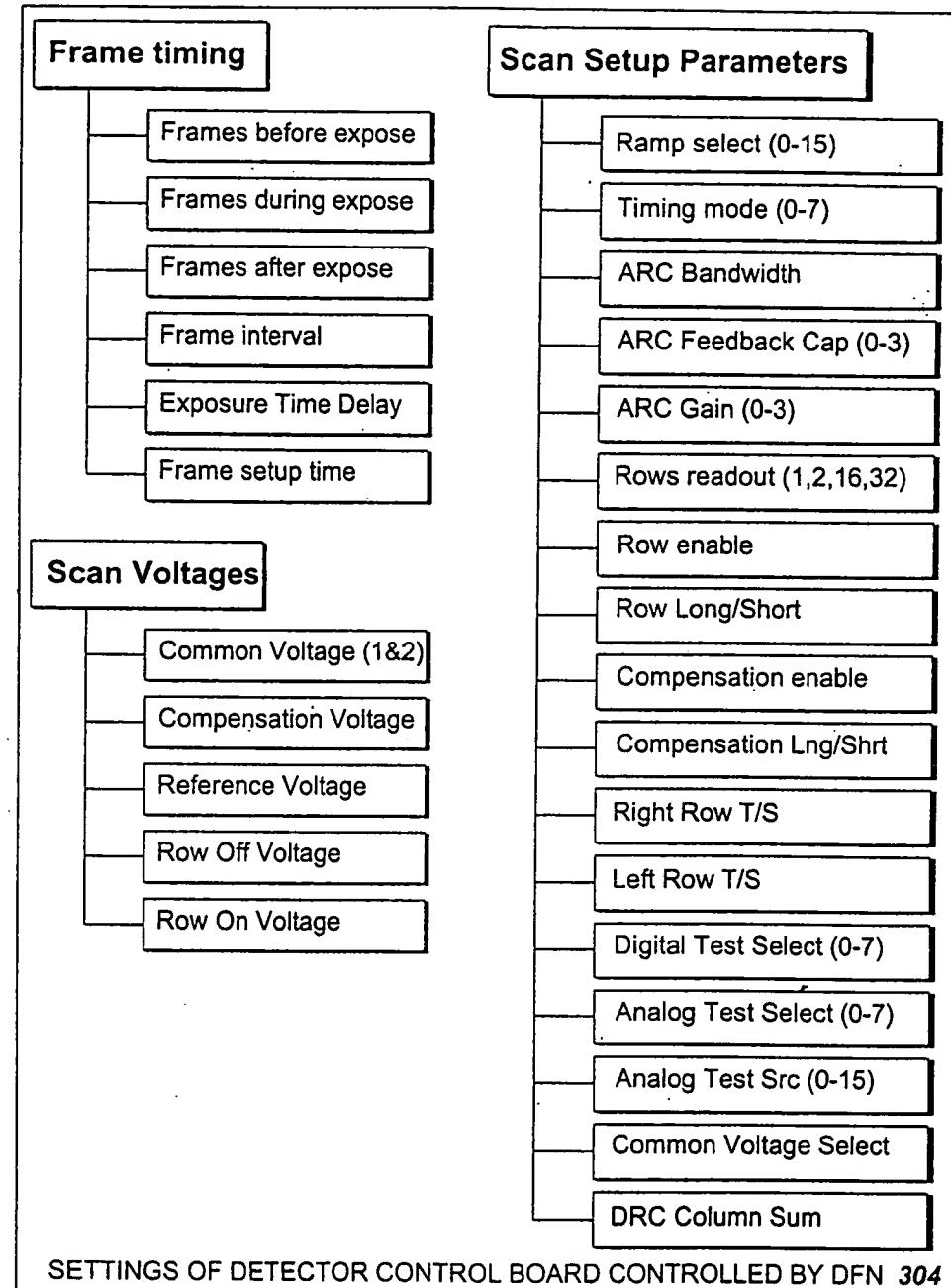
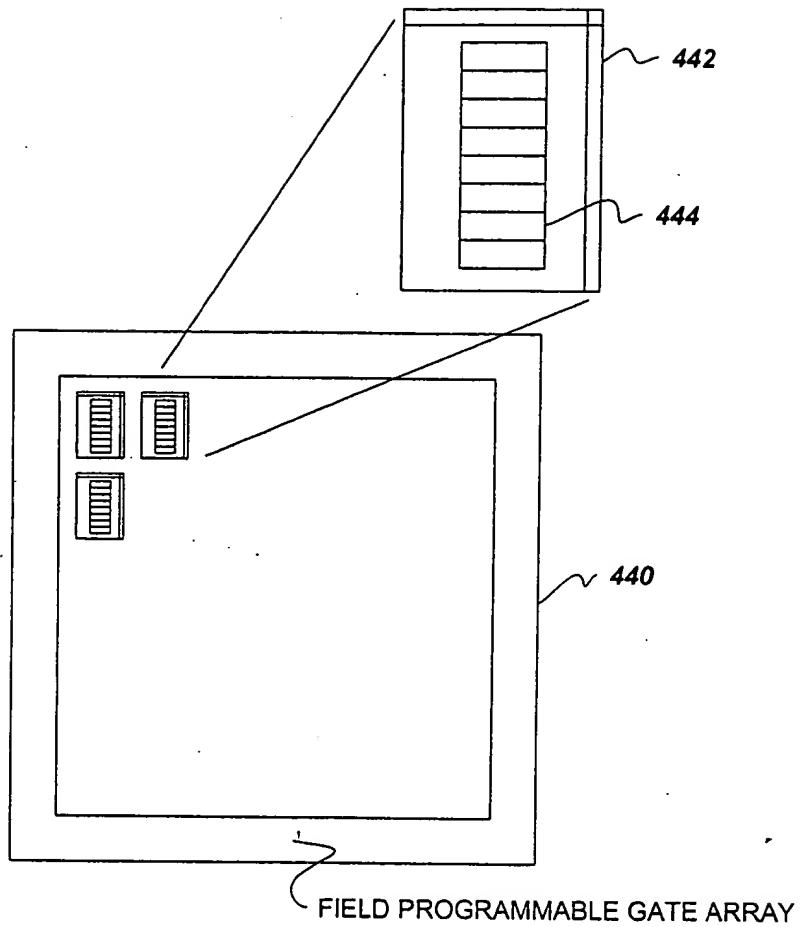


fig. 23

TOP-EDGE - BOTTOM-EDGE



FIELD PROGRAMMABLE GATE ARRAY

fig. 24

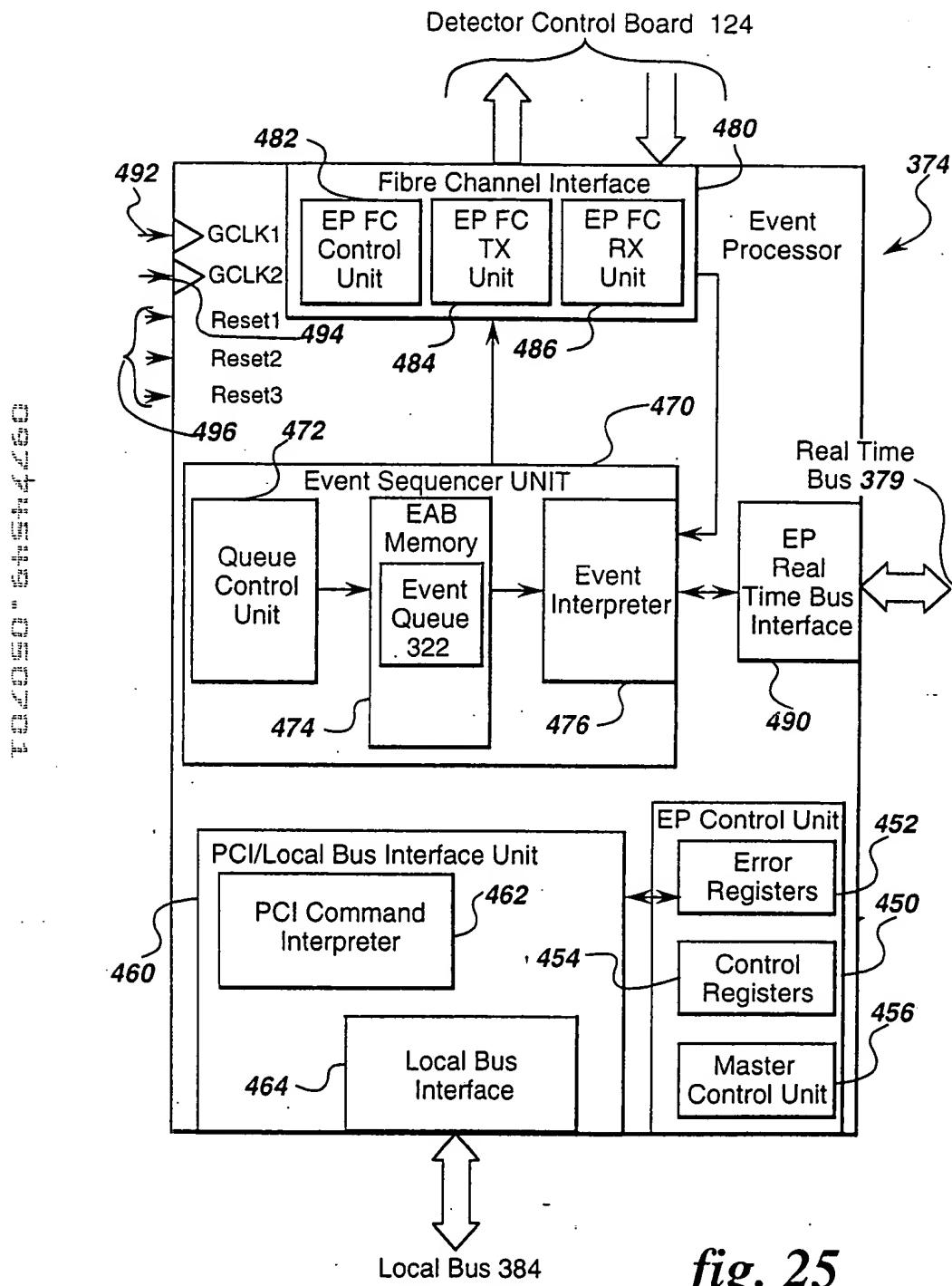


fig. 25

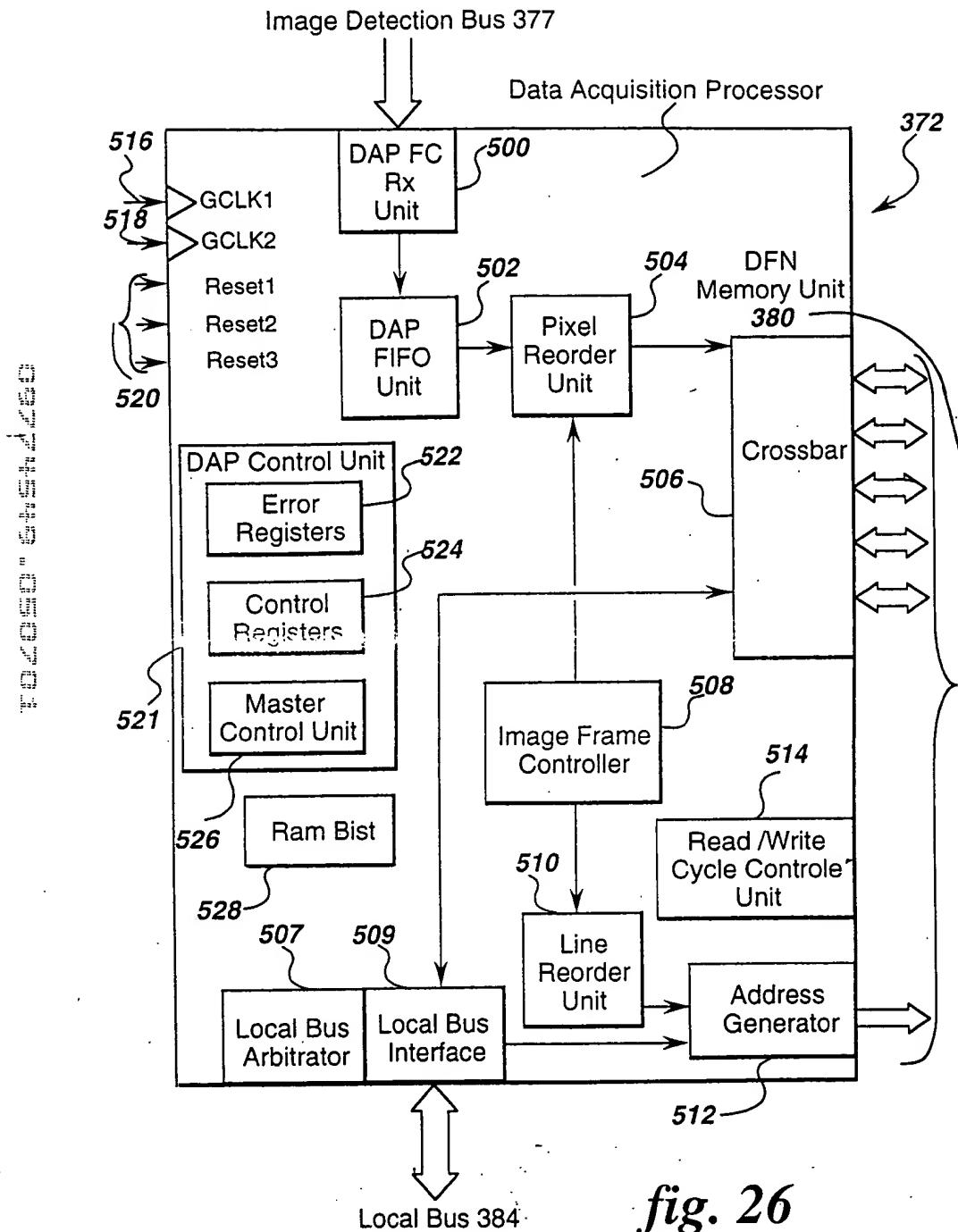


fig. 26

TOP DRAWING SHEET 260

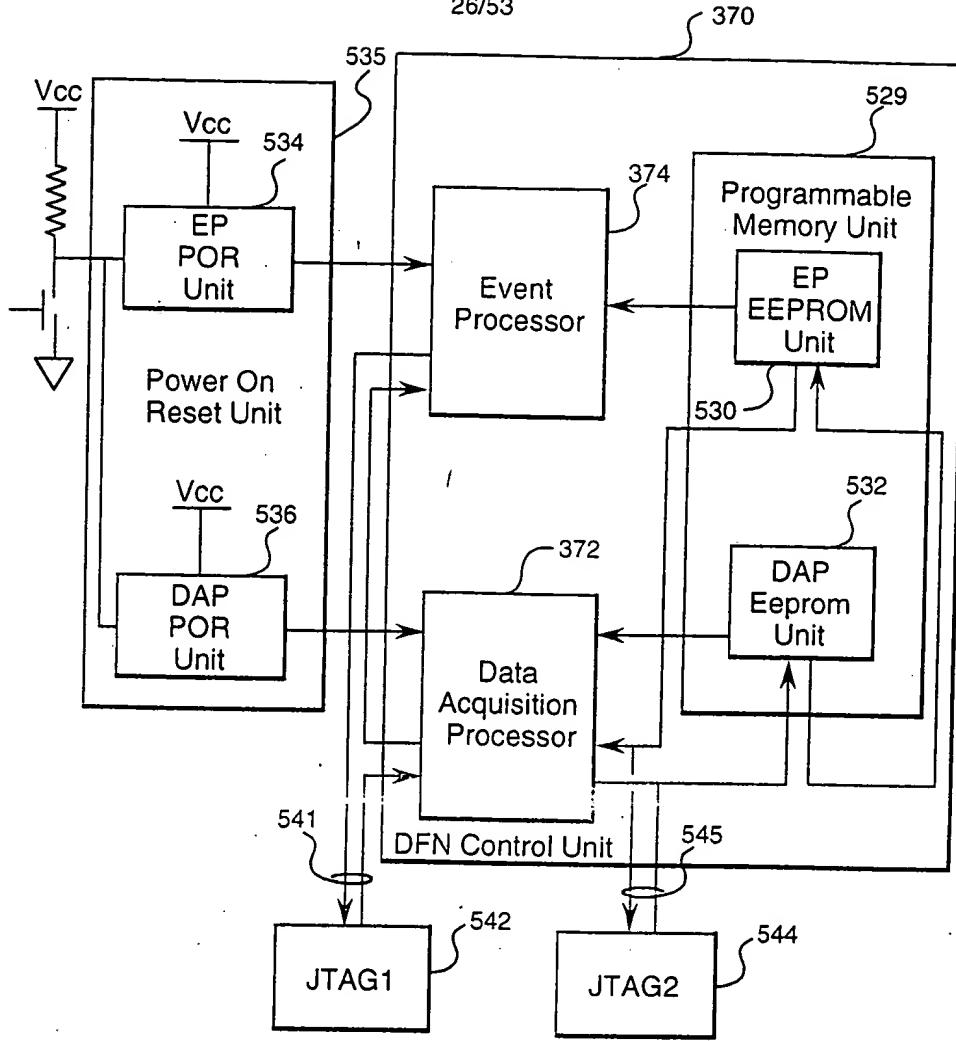


fig. 27

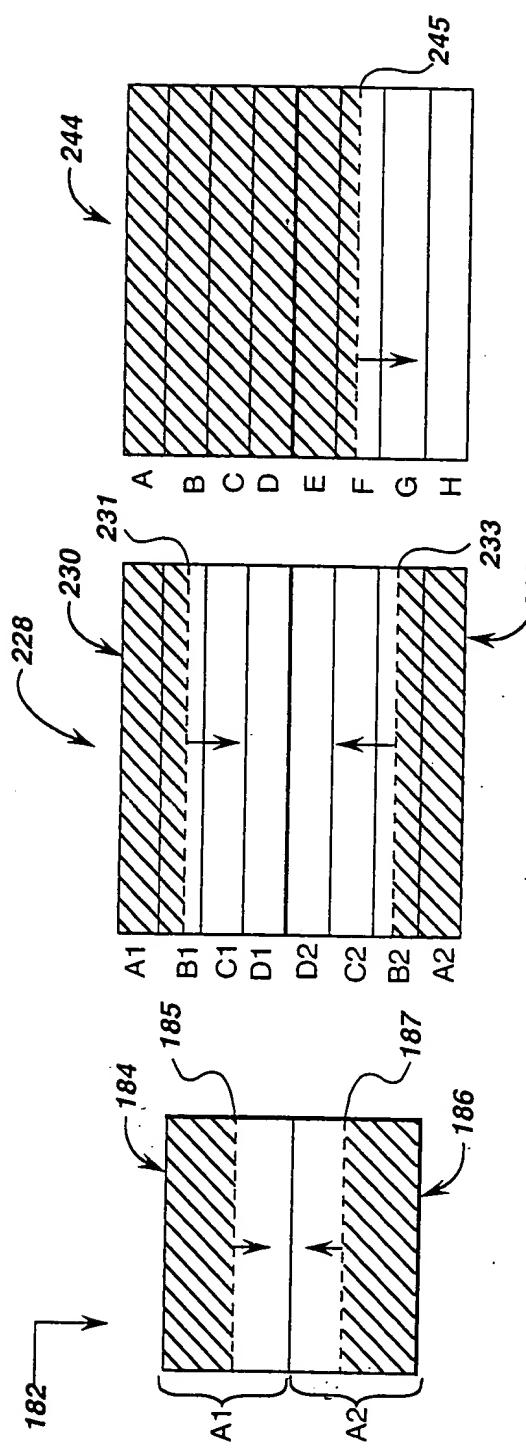


fig. 28

fig. 29

fig. 30

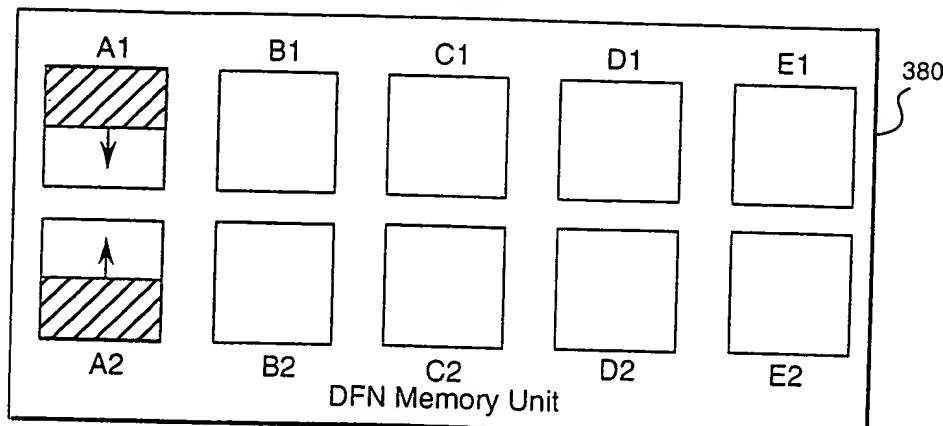


fig. 31

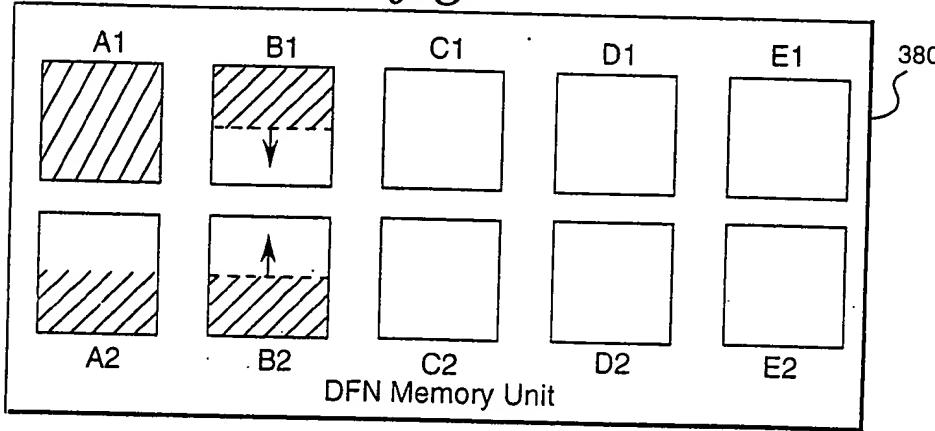


fig. 32

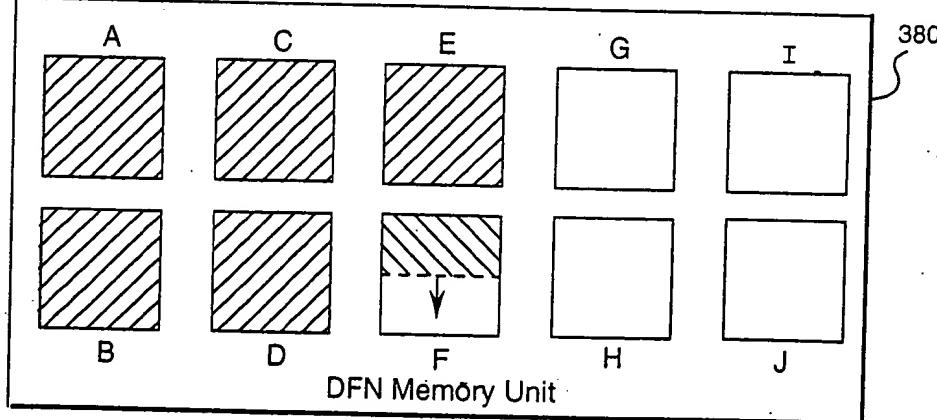


fig. 33

334

A
B
C
D
E
F
G
H

fig. 36

334

A1
B1
C1
D1
D2
C2
B2
A2

fig. 35

334

A1
A2

fig. 34

FIGURE 37: Control Card 382

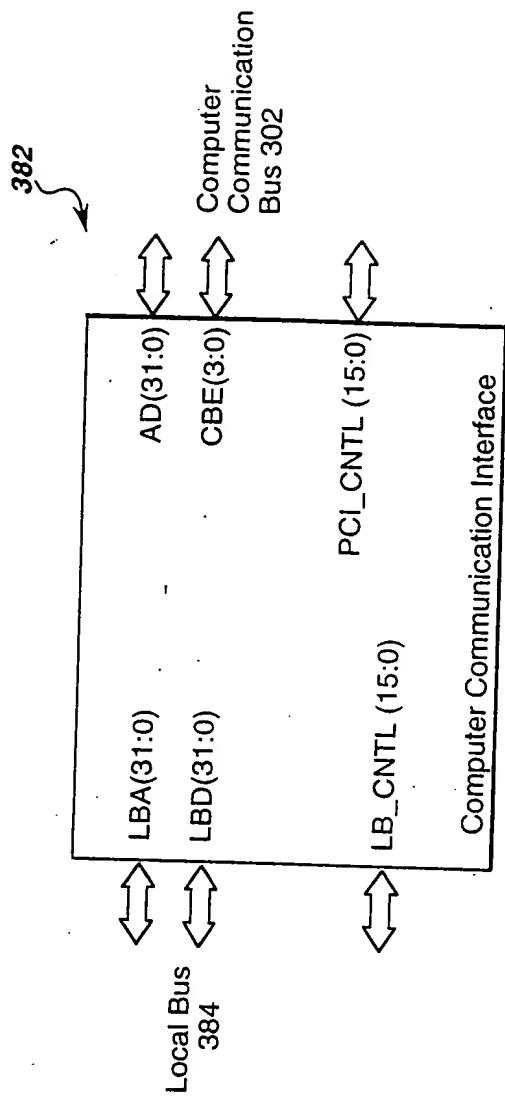


fig. 37

31/53

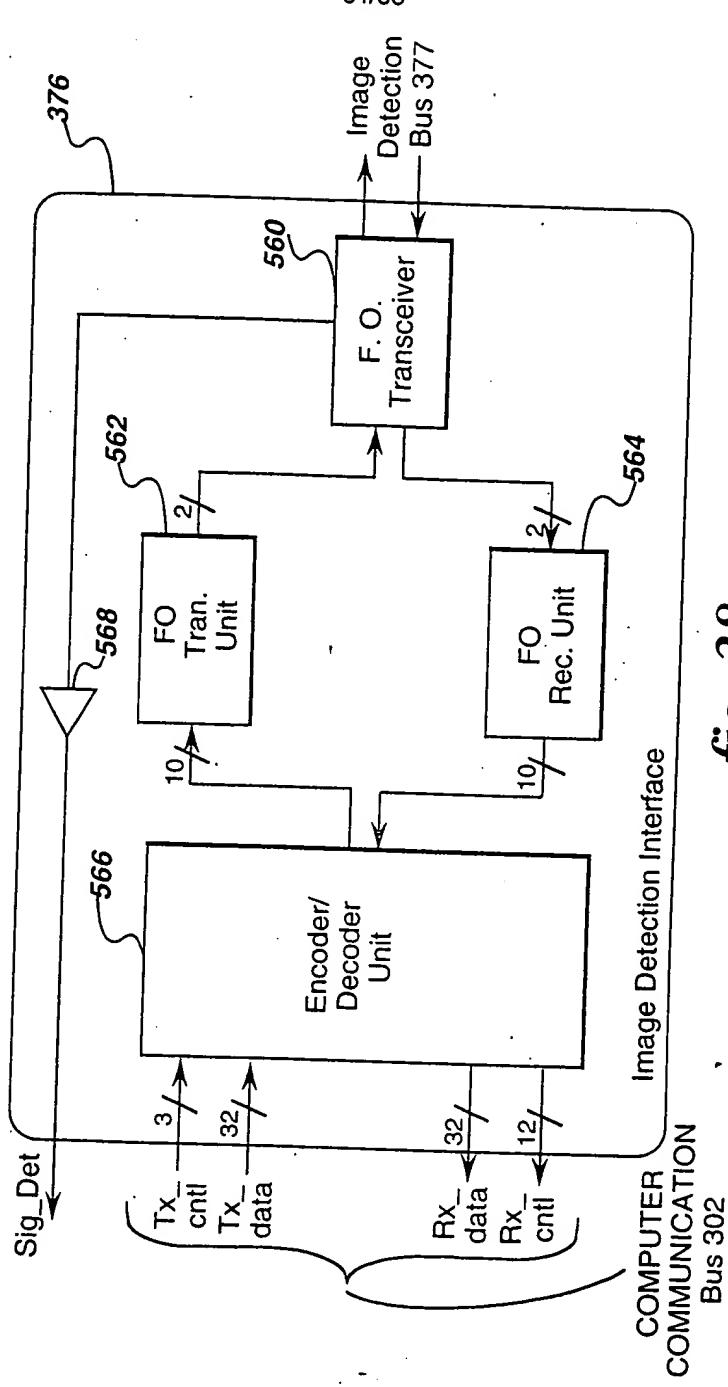


fig. 38

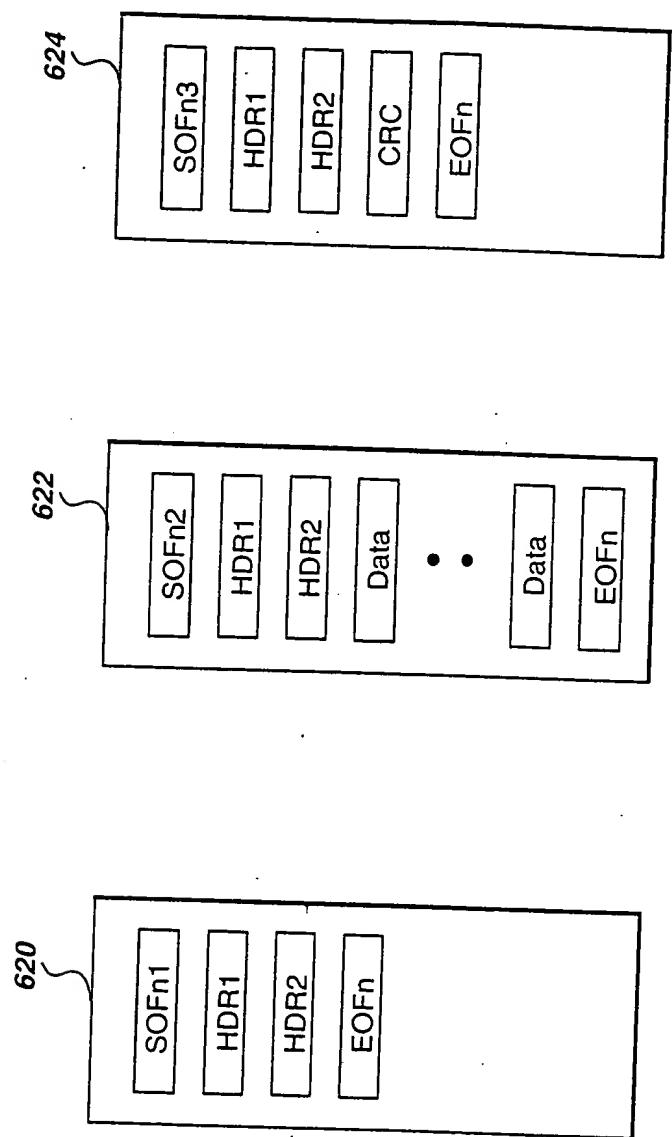


fig. 39

fig. 40

fig. 41

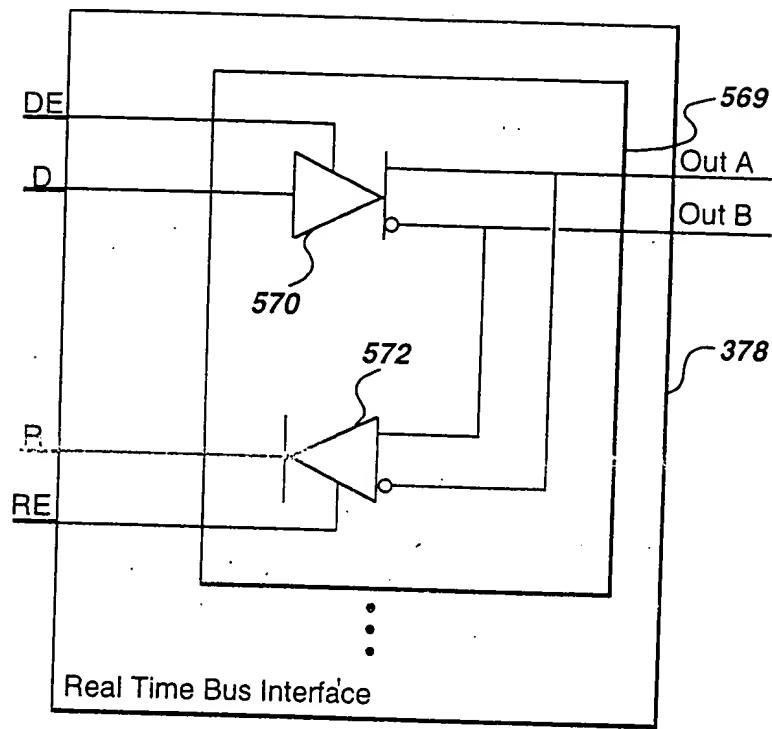
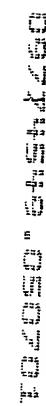
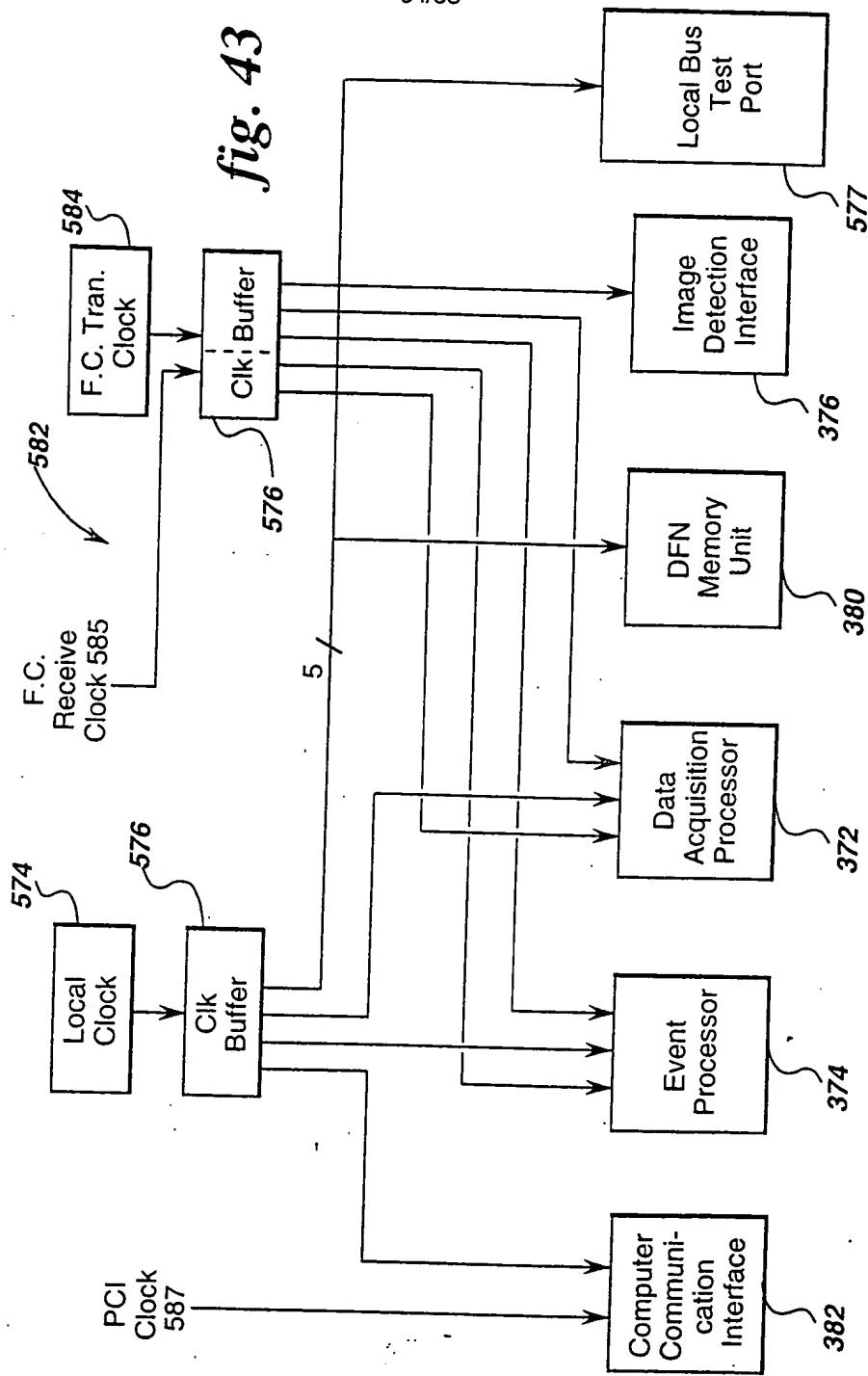
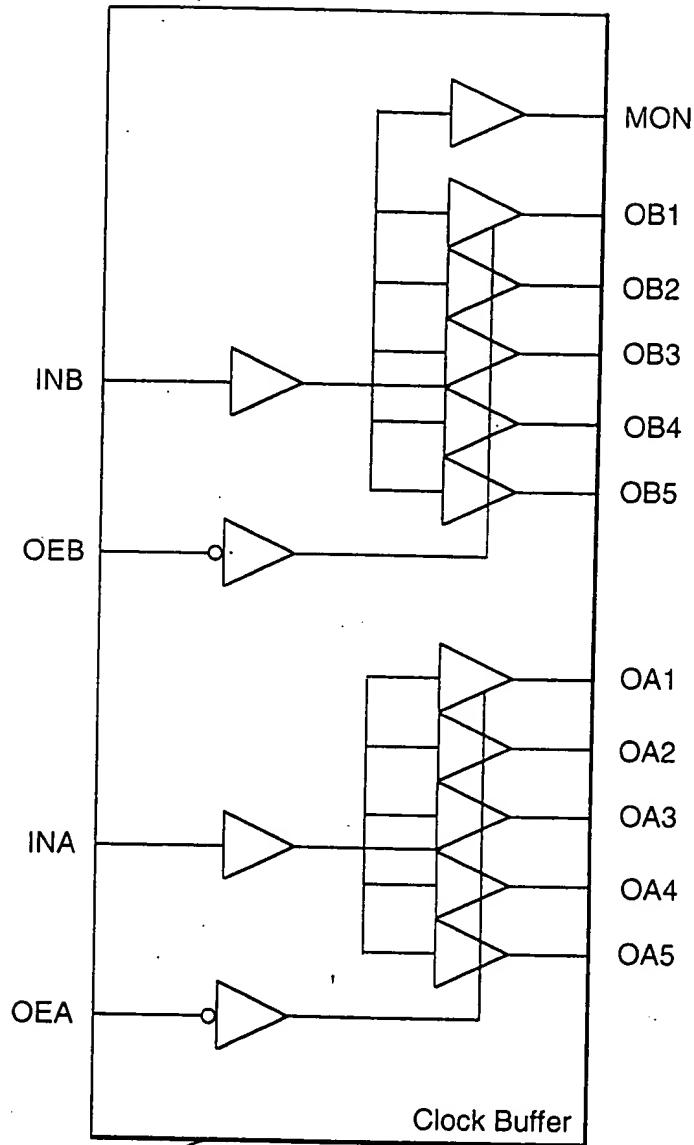


fig. 42

fig. 43



PROCESSED "SHEET 160



576

fig. 44

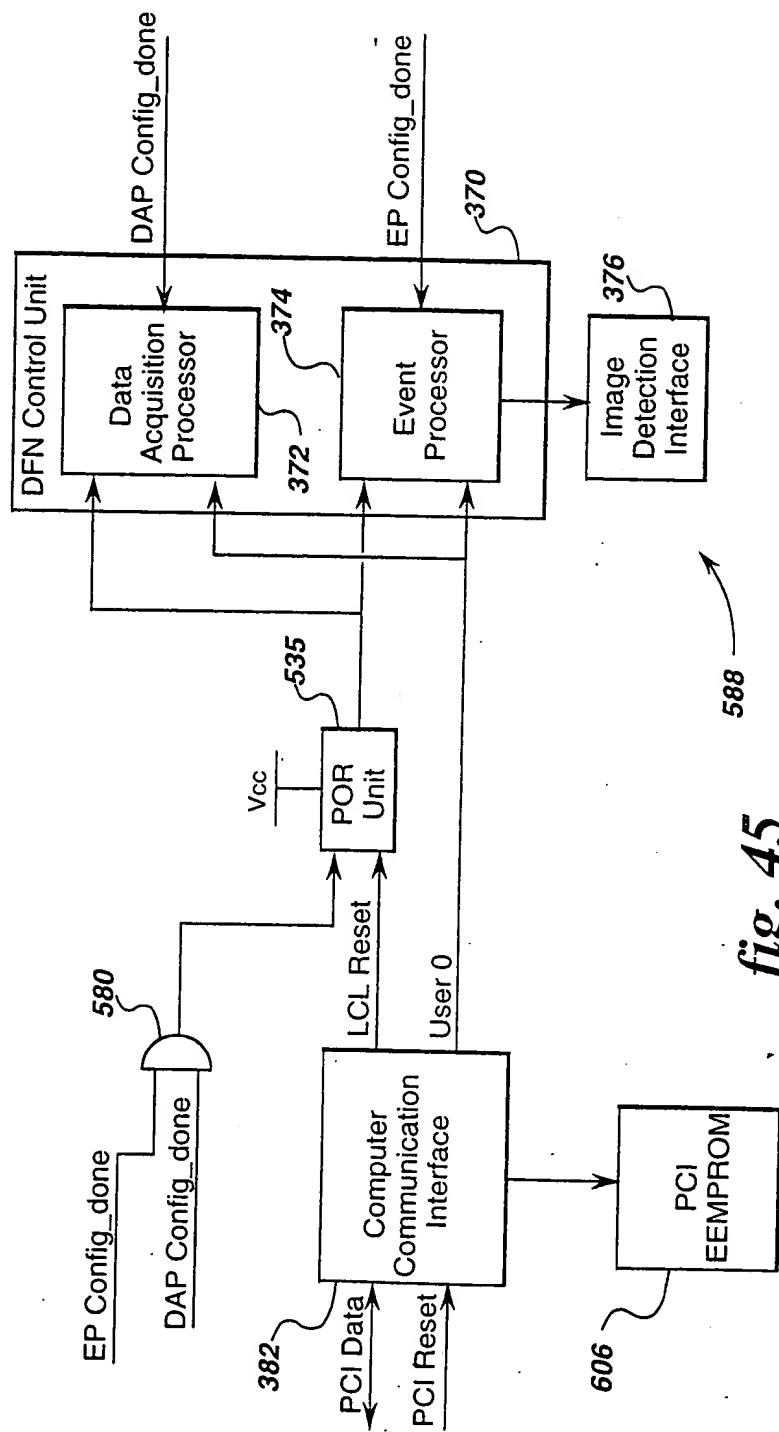


fig. 45

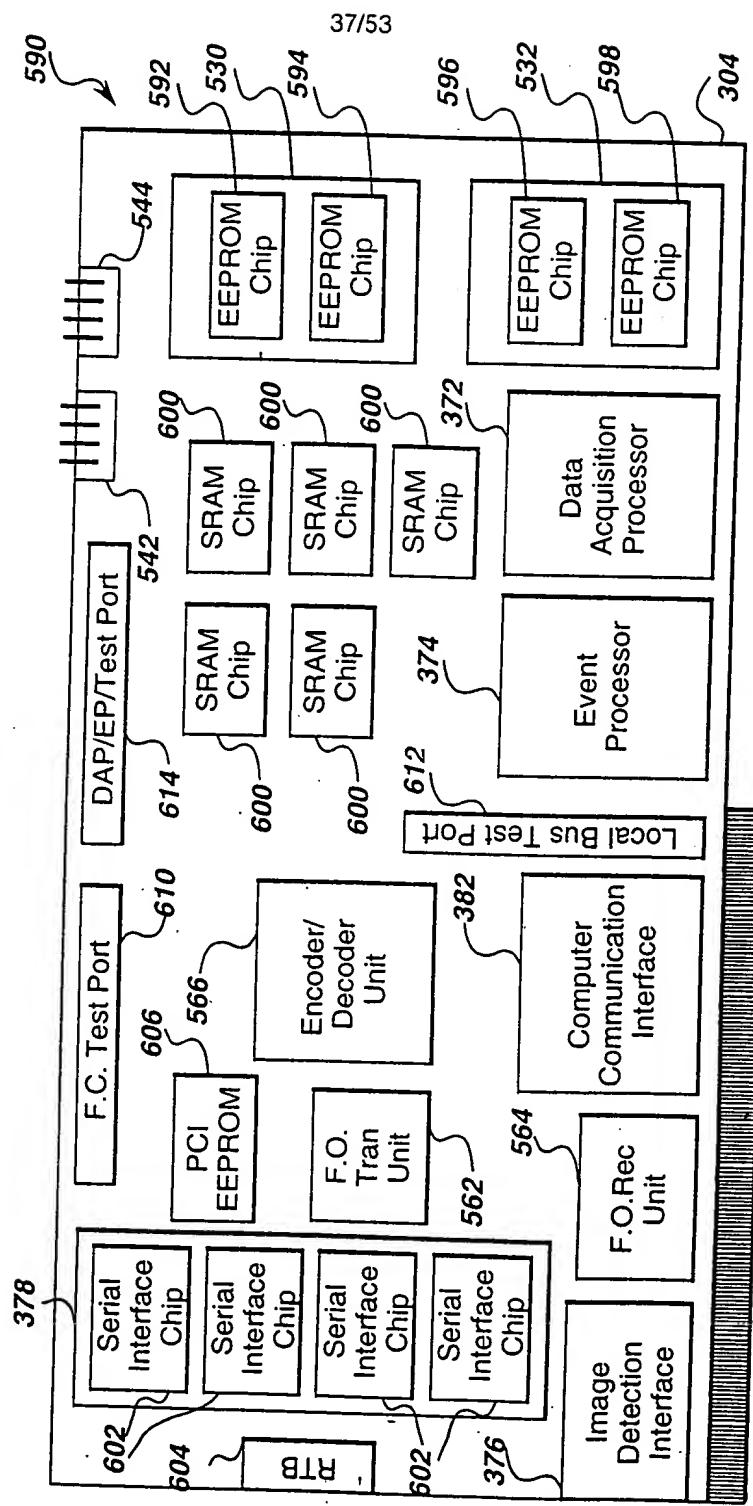
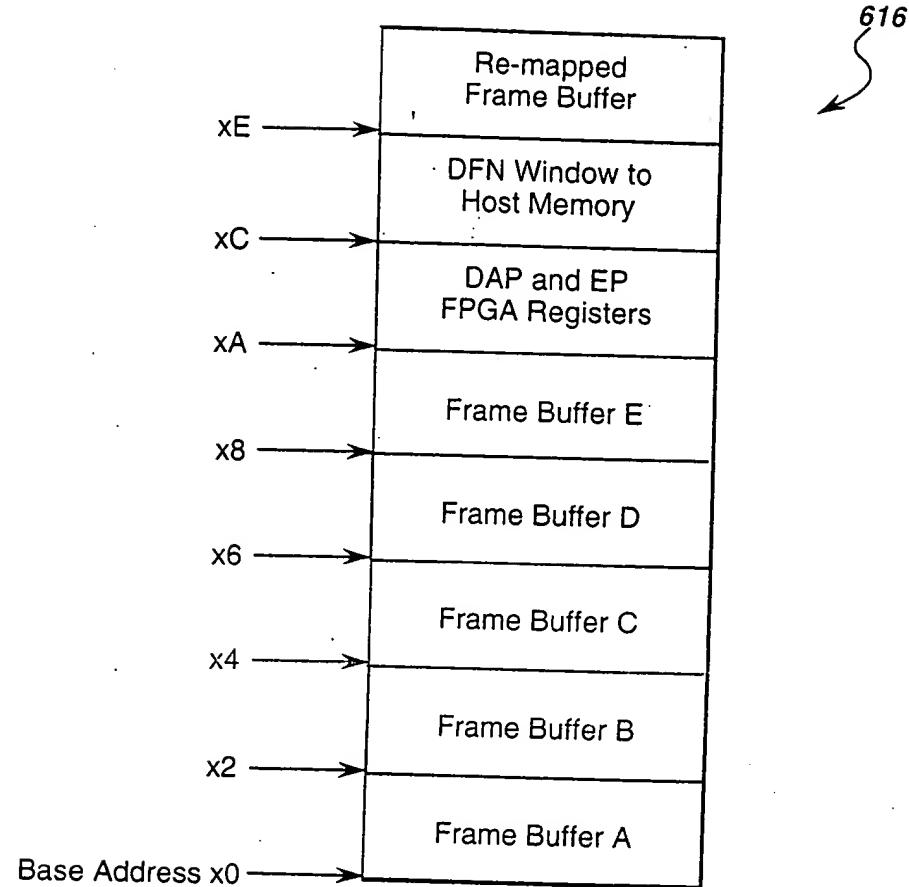


fig. 46

TOP SECRET - EYES ONLY



Mapping of 16 MByte PCI Address Space

fig. 47

PROGESS "630" TEST

fig. 48

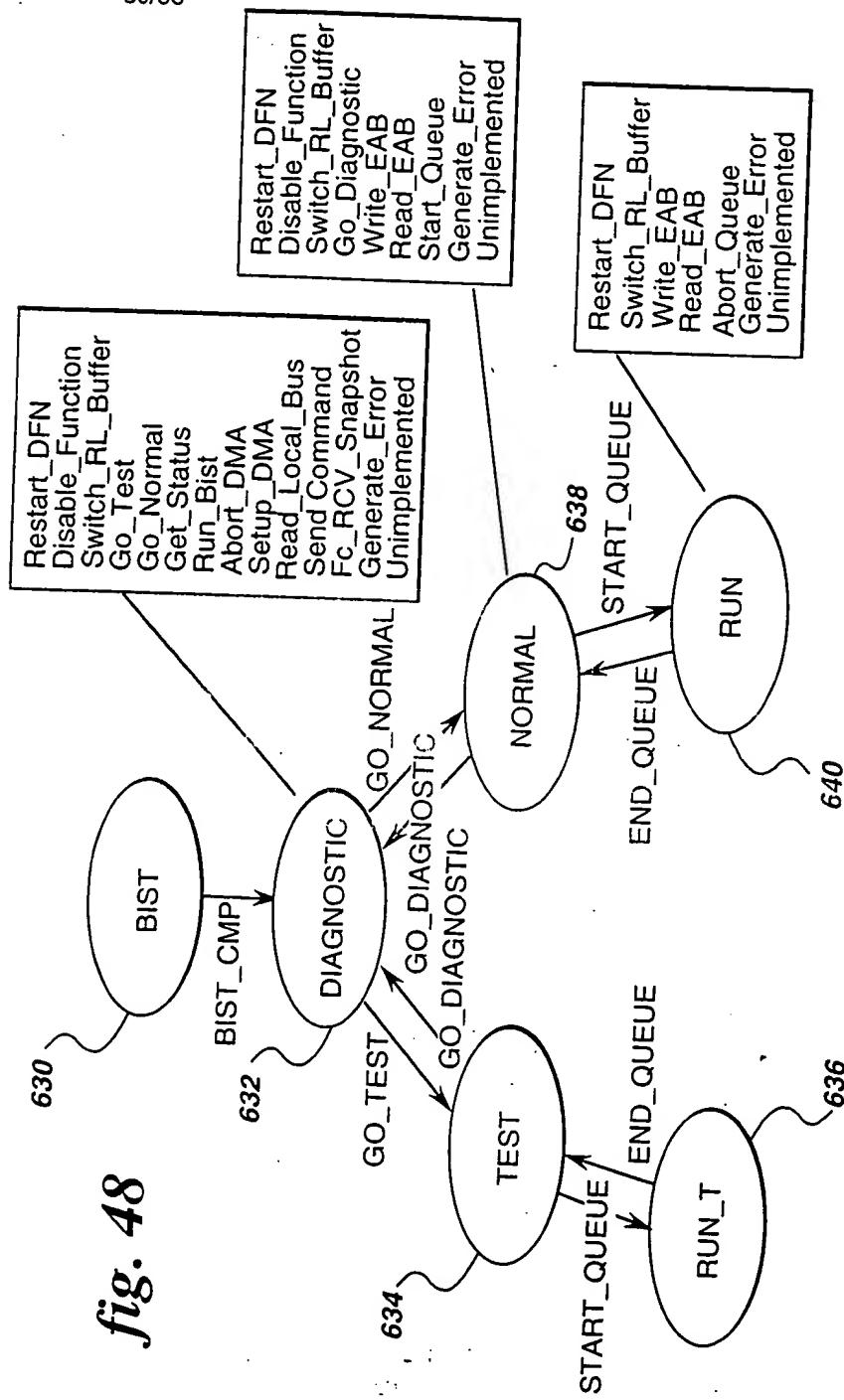


FIGURE 050 "SETUP AND SCRUB"

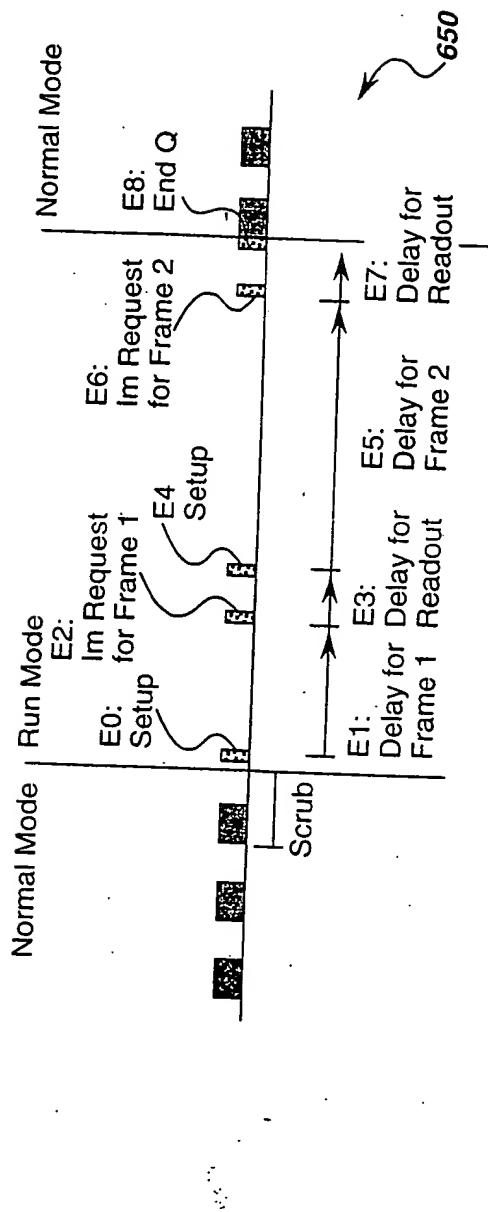


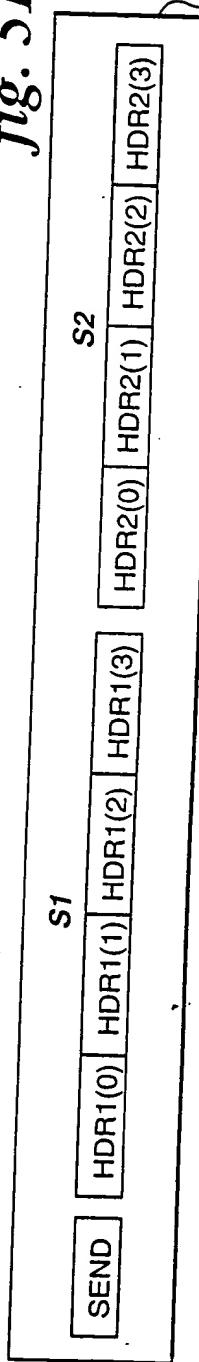
fig. 49

TOP SECRET "C445421250"

Event Mnemonic	Event (Showing size of arguments)	Op Code (hex)	Data (bytes)	Total (bytes)
Endq	Endq	14	0	1
Delay (T)	Delay (0xffff ff ff ff)	10	4	5
Send (command, value)	Send (0xffff ff ff, 0xffff ff ff)	04	8	9
LoopKN (K, N)	LoopKN (0xffff ff, 0xffff)	0C	3	4
LoopKF (K, F)	LoopKF (0xffff ff, 0xffff ff)	0D	5	6
Wait (F)	Wait (0xffff ff, ff)	09	3	4
Flag (F)	Flag (0xffff ff, ff)	08	3	4

fig. 50

fig. 51



670

Error Mnemonic	Description of Error
FC_TIMEOUT	Timeout Expired With No ACK Detected
FC_BAD_ACK	ACK Did Not Match Transmitted Command
FC_EXTRA_ACK	Unexpected ACK Received
FC_EXTRA_CMD	New Send Event While Waiting for ACK From Previous Send
SIG_DETN	No Input Signal Power on Fibre Channel (Cable Disconnected?)
RXERROR	Fibre Channel Receiver Detected Bad Data (Defective Chipset?)
WRDSYNCN	Fibre Channel Data Link Unsynchronized
CRXS(1)	Bad Received CRC Detected (Fiber-optic Cable Problem?)
CRXS(3) & CRXS(2)	Bad Order in Link State Machine (Defective Chipset?)

fig. 52

672

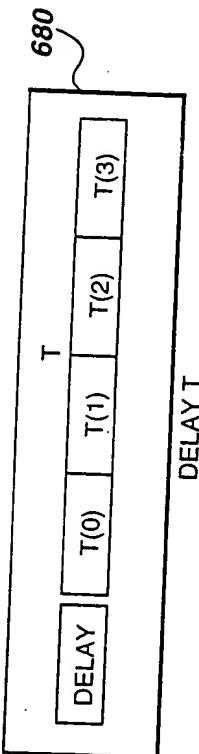


fig. 53

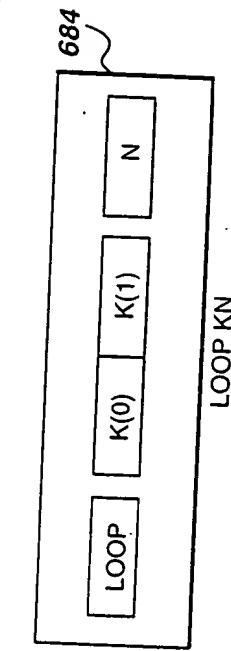


fig. 54

70240509 5th Edition 2/25/03

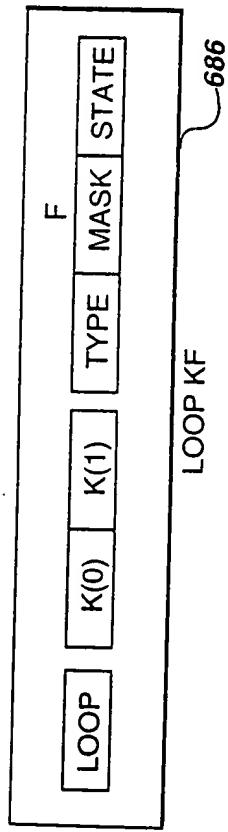


fig. 55

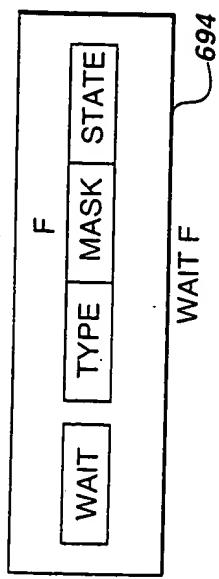


fig. 56

43/53

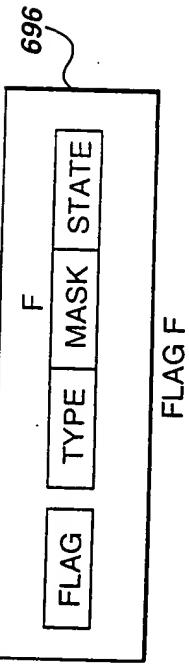


fig. 57

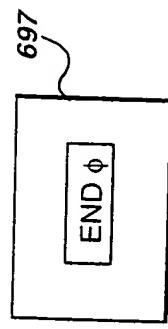


fig. 58

FIGURE 698 - ENTITLED 698

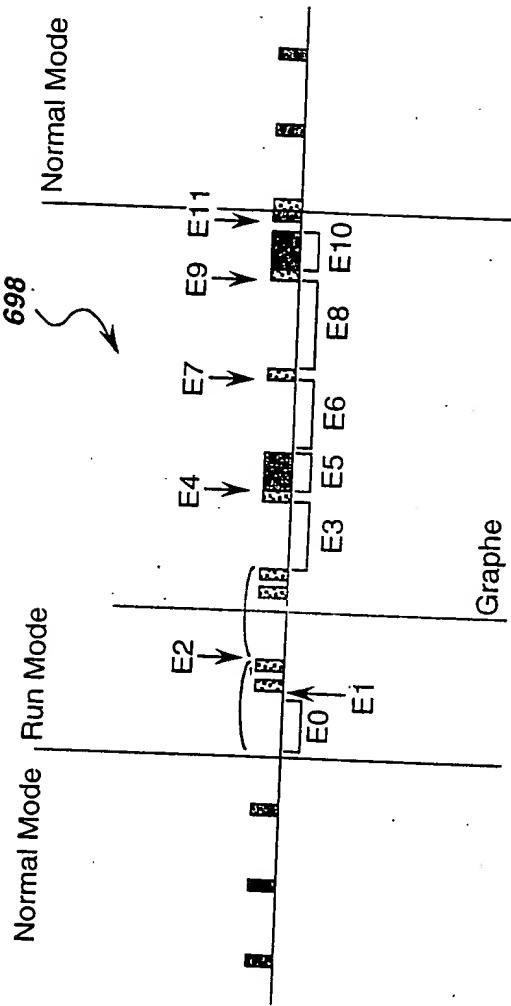


fig. 59

E11	EndQ
E10	Delay 125 ms
E9	Send Im Request
E8	Delay 500 ms
E7	Flag RT2
E6	Delay 50 ms
E5	Delay 125 ms
E4	Send Im Request
E3	Delay 300 ms
E2	Loop 2, RT1
E1	Send Scrub
E0	Delay 300 ms

EVENT QUEUE

700

fig. 60

FIGURE 61 "Format 4250"

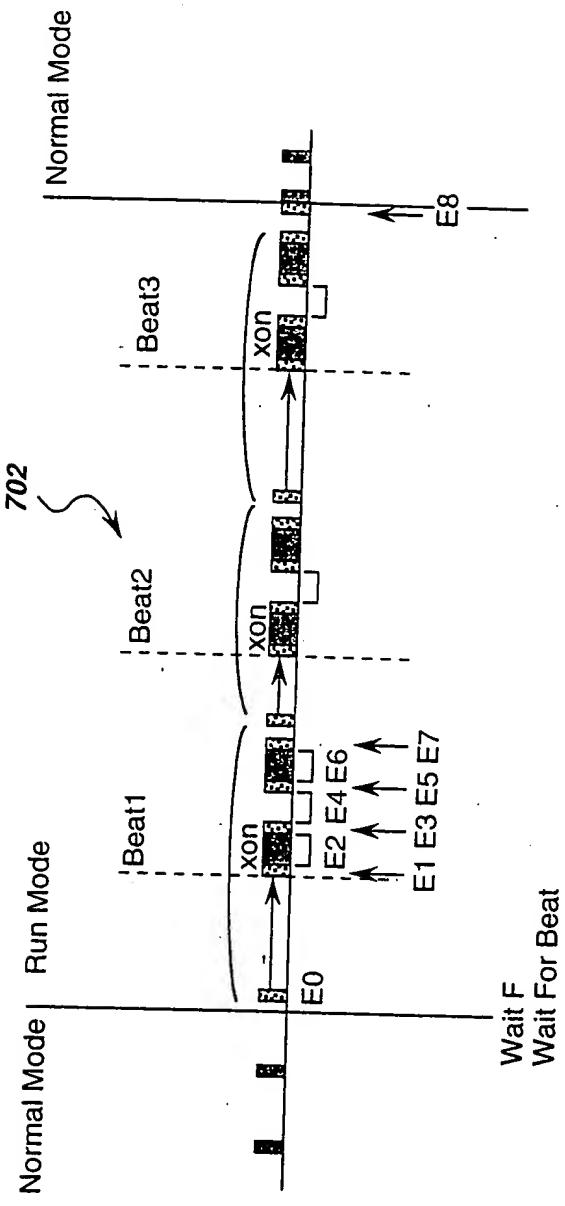
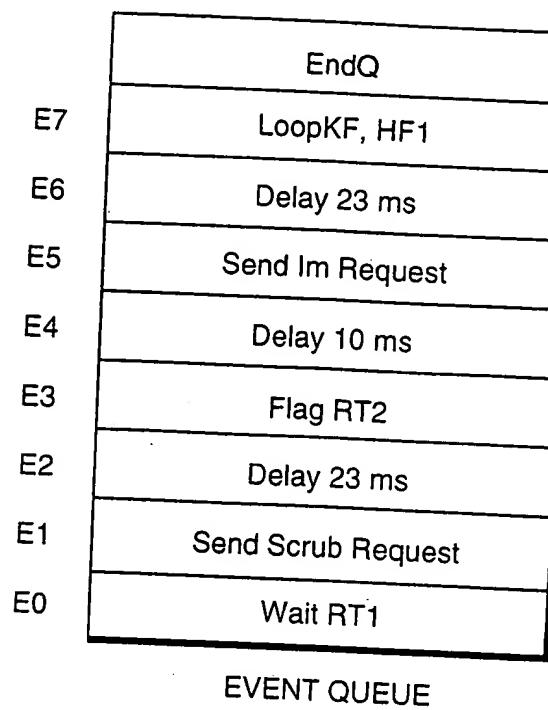


fig. 61

47/53



704
704

fig. 62

EVENT QUEUE

CONSTANT DATA

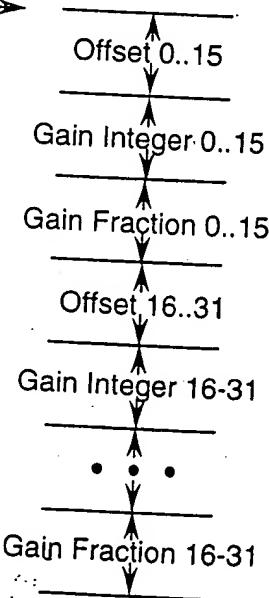


fig. 70

Constant Memory Format

TOP SECRET - EYES ONLY

TRACER 250" GENEVA 4/25/60

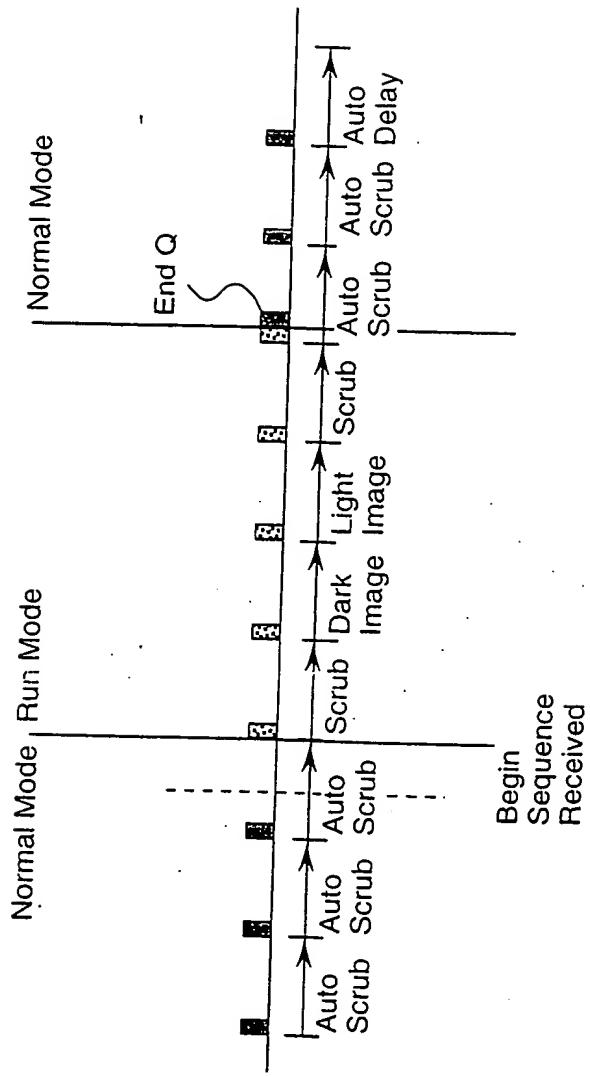


fig. 63

```
sub :frame
{
    $QVf = 'frame';
    %qv = ('delay_qv' => [10000]);
    %qp = ();
    con:file_init (@_, \%qp, \%qv, $QVf);
    Delay('Delay_qv1');
    compile_finit ();
}
```

fig. 64

```
sequence_begin ();
# define qv defaults:
%qv1 =('delay_qv' => 5000);
# call frame with qv's
frame_type1(NULL, \%qv1, 1);
sequence_end ();
```

fig. 65

```
pDFN->DFNChangeQueueVariable
(
    (char *) SymName,
    (char *) sndBuf,
    BufSize
    (ULONG *) & Debug
);
```

// variable name
// new value
// num bytes to write
// developer info

fig. 66

FIGURE 67 "GETTING STARTED"

```
// load and run the event sequence
pDFNBeginSequenceNoMappingNoLog
  (snm, "d:\Hf.bin");
// assign data to be passed
sndBuf = 25000;
// change the queue variable
pDFN->DFNChange QueueVariable
  (
    (char *) SymName,           // variable name
    (char *) sndBuf,           // new value
    (ULONG ) sizeof sndBuf,    // num bytes to write
    (ULONG *) & debug,          // developer info
  );
```

```
sub frame_type1
  {
    $Hfrm = 'frame_type1';
    %qv = ('delay_qv' => [20000]);
    %qp = ( );
    $image_cmd = [0x800000,0x0];
    compile_init (@_, \%qp, \%qv, $Hfrm);
    Send ($image_cmd);
    Delay('delay_qv');
    LoopKF(2, 0xAFF01);
    compile_fini( );
  }
```

fig. 67

fig. 68

TRADE SHOW GRAPHICS

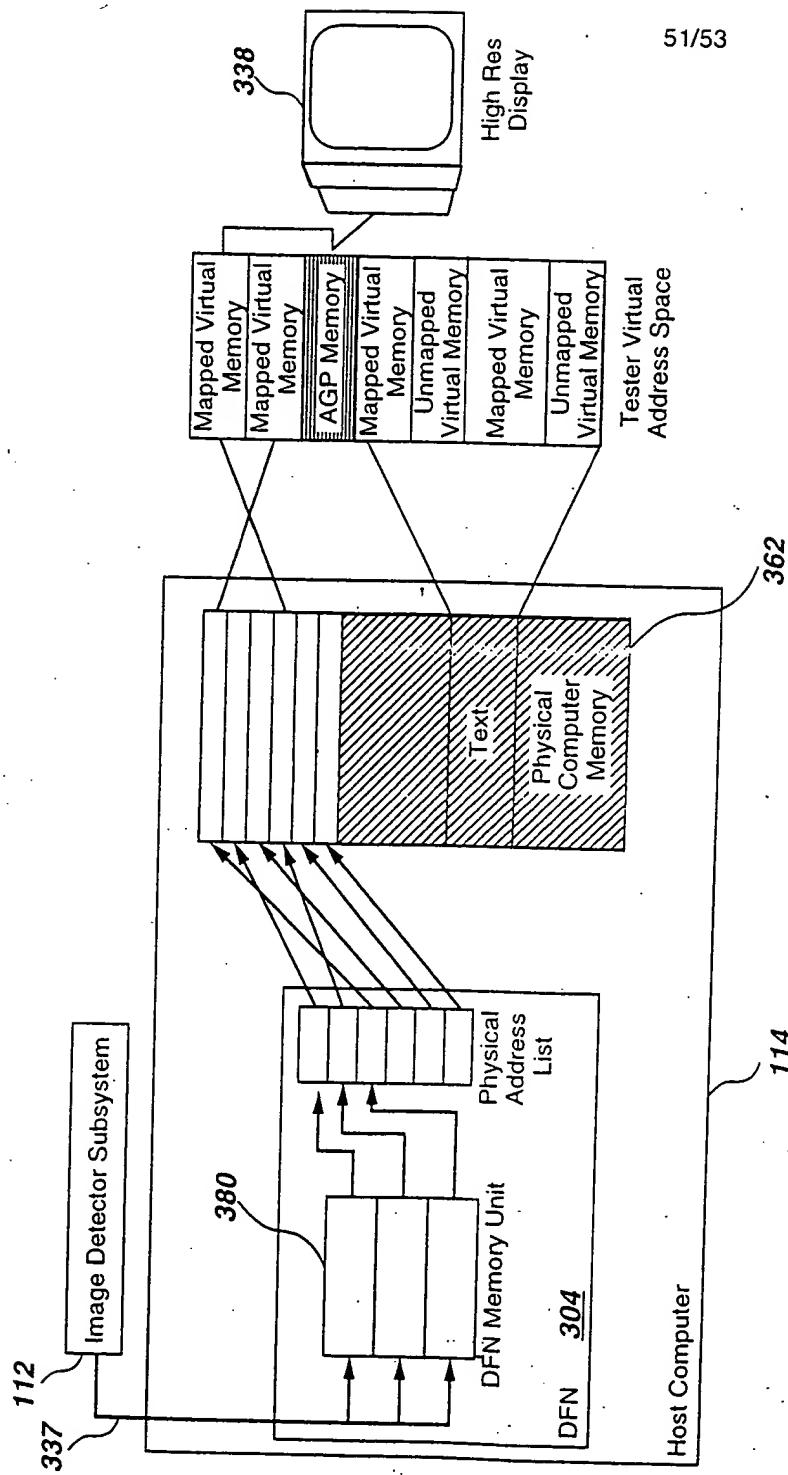


fig. 69

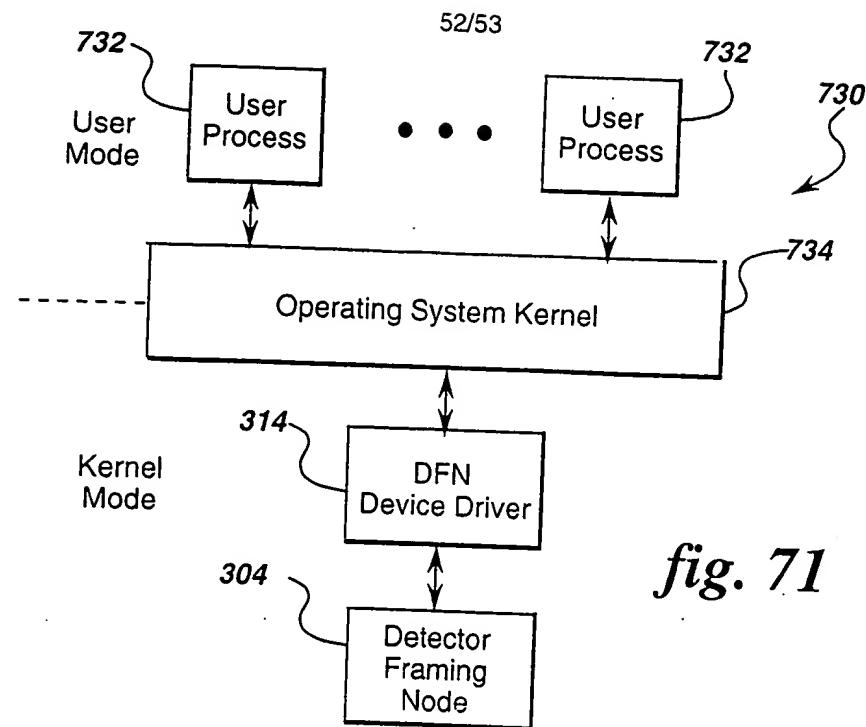


fig. 71

TOP SECRET - 025054259

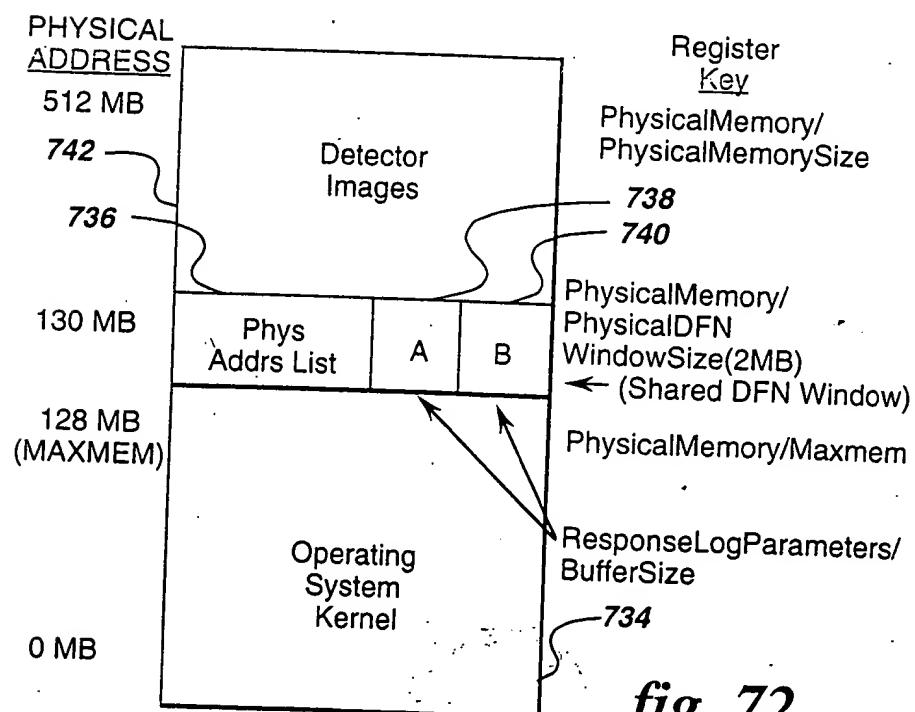


fig. 72

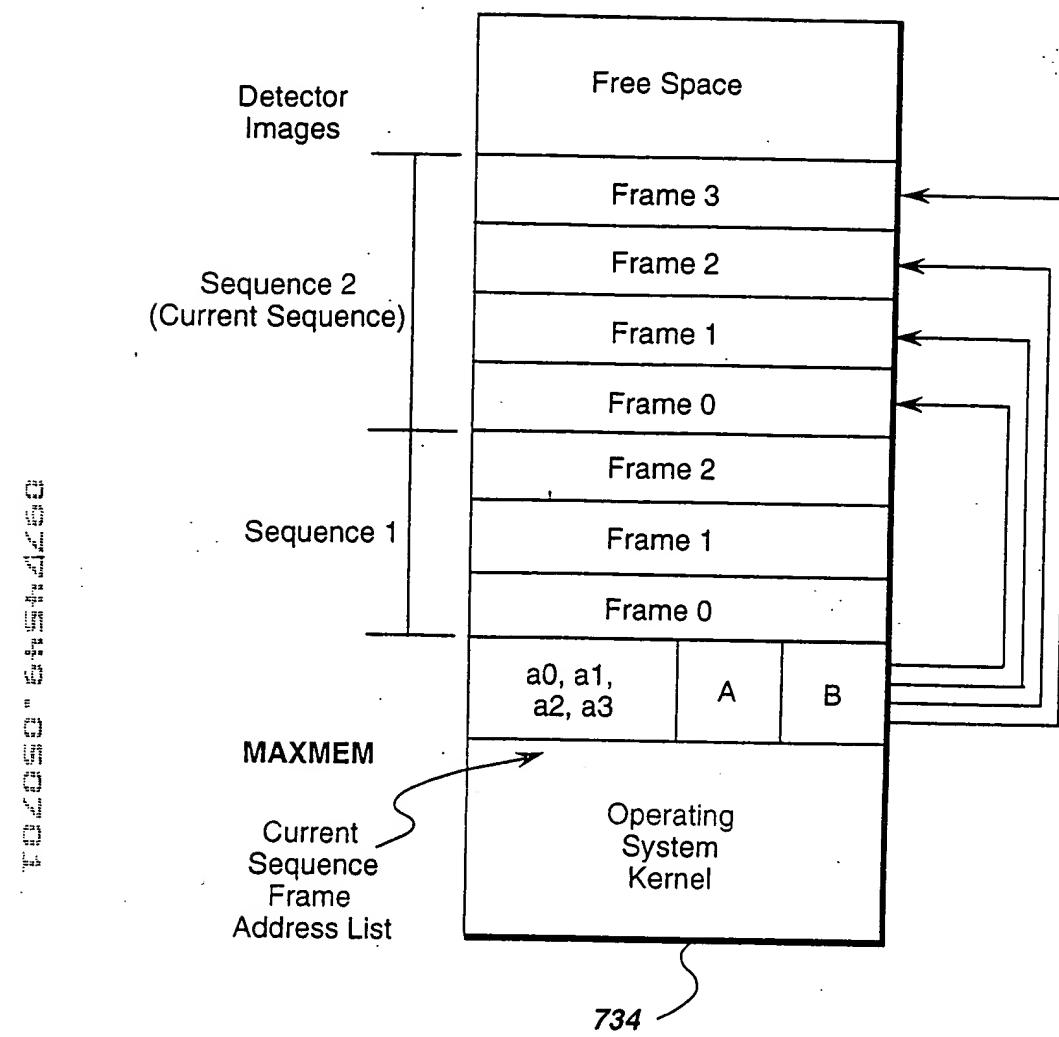


fig. 73